

UTILIZAÇÃO DA TECNOLOGIA BIM NA CARACTERIZAÇÃO DO BETÃO EM ESTRUTURAS EXISTENTES

JOANA PATRÍCIA MARTINS PINTO

Dissertação submetida para satisfação parcial dos requisitos do grau de
**MESTRE EM ENGENHARIA CIVIL — ESPECIALIZAÇÃO EM MATERIAIS E PROCESSOS DE
CONSTRUÇÃO**

Professor Doutor Miguel Ângelo Carvalho Ferraz

JUNHO DE 2018

MESTRADO INTEGRADO EM ENGENHARIA CIVIL 2012/2013

DEPARTAMENTO DE ENGENHARIA CIVIL

Tel. +351-22-508 1901

Fax +351-22-508 1446



miec@fe.up.pt

Editado por

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Rua Dr. Roberto Frias

4200-465 PORTO

Portugal

Tel. +351-22-508 1400

Fax +351-22-508 1440



feup@fe.up.pt



<http://www.fe.up.pt>

Reproduções parciais deste documento serão autorizadas na condição que seja mencionado o Autor e feita referência a *Mestrado Integrado em Engenharia Civil - 2014/2015 - Departamento de Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Porto, Portugal, 2015.*

As opiniões e informações incluídas neste documento representam unicamente o ponto de vista do respetivo Autor, não podendo o Editor aceitar qualquer responsabilidade legal ou outra em relação a erros ou omissões que possam existir.

Este documento foi produzido a partir de versão eletrónica fornecida pelo respetivo Autor.

Aos meus Pais e Avós

Even if you are on the right track, you will get run over if you just sit there

Will Rogers

AGRADECIMENTOS

A materialização deste trabalho contou com a dedicação pessoal, mas também com o apoio de algumas pessoas que, de modos diferentes, contribuíram para o meu percurso até aqui. Aproveito este espaço para refletir sobre o papel de cada um e tentar espelhar, da melhor forma que me é possível, a sincera gratidão que sinto por cada um deles.

Ao meu orientador, o Professor Miguel Ferraz, pela oportunidade. Por aceitar temas inovadores, apostando na ideia base deste projeto desde o primeiro momento que lhe foi proposto. Pela constante disponibilidade, compreensão e sensibilidade demonstradas em todo o processo, alertando para as dificuldades e para o risco em torno deste tema. O meus mais sinceros agradecimentos.

Aos meus pais, João e Maria, por investirem na minha formação académica, por acreditarem nas minhas capacidades e por tanto contribuírem para o meu desenvolvimento enquanto pessoa. Não há agradecimento suficiente que iguale as oportunidades, o saber e as realizações que eles tornam possíveis.

Ao Luís, por me acompanhar há tantos anos como amigo, companheiro e professor que, com um sorriso sempre persistente, encontra uma solução para todos os meus dilemas. Ao seu companheirismo, à sua paciência e à sua perseverança. Reconheço que grande mérito desta dissertação se deve à sua ajuda incondicional, na resolução de erros de código, e às muitas tardes a trabalhar em conjunto, à procura de melhores soluções.

Ao meu amigo Fábio Dinis, pela partilha constante de conhecimento e pela total disponibilidade. Por me levar por novos caminhos, não só durante este trabalho, mas ao longo destes cinco anos de amizade.

Às minhas irmãs, Marta e Beatriz, pela paciência demonstrada nestas últimas semanas, pelas palavras de carinho e pelos passeios noturnos quando o stress de um dia de trabalho ditava as leis do relacionamento.

Aos meus amigos, Maria José, Natália, Alberto e Jorge, pela disponibilidade total, pela compreensão do esforço envolvido e pelos almoços e jantares de convívio que, em muito contribuíram para a minha “sanidade mental”. Sobre tudo, pelos conselhos experientes que irei acumulando ao longo a vida.

Ao Luís Sanhudo, por estar disponível a ajudar, tirando um pouco do seu tempo para aliviar o stress de um desconhecido, e por me elucidar para as artes da programação visual em tão pouco tempo.

RESUMO

Com o aumento do interesse em obras de manutenção e reabilitação, a classificação dos materiais existentes é uma das avaliações necessárias para assegurar o bom funcionamento da estrutura. O betão é um dos materiais mais utilizados, pelo que se recorre constantemente a ensaios, destrutivos e não-destrutivos, para a sua caracterização.

Deste modo, uma nova abordagem ao processo de classificação do betão consiste na automatização dos métodos normalizados em detrimento da abordagem manual utilizada convencionalmente. Surge, então, a ideia de aliar aos processos e matérias de construção à metodologia BIM.

Nesta dissertação foram desenvolvidas aplicações informáticas que permitem, a um utilizador, tirar partido da modelação paramétrica do BIM para identificar áreas de ensaio, armazenar os dados obtidos e classificar o betão. Estas aplicações permitem obter a classe de resistência do betão, não só em obras já existentes, mas também, em novas obras.

Para a elaboração deste documento foram utilizadas as normas NP EN 13791 e EN 12504-2. A primeira refere-se à avaliação da resistência à compressão do betão nas estruturas e em produtos pré-fabricados. A segunda está relacionada com ensaios de betão nas estruturas, mais concretamente na determinação do índice esclerométrico. Como tecnologias da informação foram utilizadas ferramentas BIM, como o *Revit*, para modelação de uma estrutura real, mas também linguagens de programação como o *Python* e o *Dynamo*, para criação de um *plug-In*.

PALAVRAS-CHAVE: Materiais, Betão, Automatização, BIM, *Plug-In*.

ABSTRACT

With the increasing number of maintenance and rehabilitation works, the classification of existing materials is one of the necessary evaluations to ensure the proper functioning of a structure. Concrete is one of the most commonly used materials, and therefore, destructive and non-destructive tests are used for its characterization.

Thus, a new approach to concrete classification process is the automation of standardized methods over the conventionally used manual approach. Then comes the idea of combining the processes and building materials to BIM methodology.

In this dissertation, software applications were developed in order to allow a user to take advantage of BIM's parametric modelling which purpose is to identify test areas, store the obtained data and classify concrete. These applications allow to obtain the concrete's resistance class, not only in existing construction, but also, in new constructions.

For the elaboration of this document, NP EN 13791 and EN 12504 were used. The first one refers to the assessment of in-situ compressive strength in structures and precast concrete components. The second is related to concrete tests in structures, more specifically, determination of rebound number. BIM tools, such as Revit, were used as information technologies in order to model a real structure. Programming languages, such as Python and Dynamo, were also used to create a Plug-in.

KEYWORDS: Materials, concrete, automatics, BIM, Plug-In

ÍNDICE GERAL

AGRADECIMENTOS	i
RESUMO	iii
ABSTRACT	v
 1. INTRODUÇÃO	 1
1.1. ÂMBITO	1
1.2. MOTIVAÇÃO	2
1.3. OBJETIVOS	2
1.4. ESTRUTURA DA DISSERTAÇÃO	3
 2. ESTADO DA ARTE	 5
2.1. BUILDING INFORMATION MODELING – BIM	5
2.1.1. BREVE HISTÓRIA	5
2.1.2. DEFINIÇÃO	6
2.1.3. VANTAGENS E DESVANTAGENS	8
2.1.4. DIMENSÕES DOS MODELOS	12
2.1.5. NÍVEIS DOS MODELOS BIM	13
2.1.5.1. Níveis de Maturidade	13
2.1.5.1. Níveis de Desenvolvimento	14
2.2. MATERIAIS COM INFORMAÇÃO BIM	16
 3. CONSTRUÇÃO DO MODELO	 21
3.1. MODELAÇÃO DA PILARES E PAREDES	22
3.2. MODELAÇÃO VIGAS	25
3.3. MODELAÇÃO DE LAJES	27
3.3. MODELAÇÃO DE CAROTES E PONTOS DE LEITURA DO ESCLERÓMETRO	28
 4. CRIAÇÃO DO PLUG-IN	 29
4.1. CONSTRUÇÃO DO ALGORITMO	30
4.2. INICIAÇÃO AO C#	32

4.3. INICIAÇÃO AO DYNAMO	35
4.4. ESCOLHA DO MÉTODO	36
4.5. PROGRAMAÇÃO DO PLUG-IN	37
4.6. DIAGRAMA DE UTILIZAÇÃO DO PLUG-IN	43
5. CASO DE ESTUDO	47
5.1. DESCRIÇÃO	47
5.2. UTILIZAÇÃO DO PLUG-IN	49
6. CONCLUSÕES E DESENVOLVIMENTOS FUTUROS	53
6.1. CONCLUSÃO	53
6.2. DESENVOLVIMENTOS FUTUROS	54
BIBLIOGRAFIA	55
ANEXOS	59
ANEXO 1	61
ANEXO 2	63
ANEXO 3	65
ANEXO 4	67
ANEXO 5	69
ANEXO 6	71

ÍNDICE DE FIGURAS

Figura 1 - BIM no ciclo de vida de uma estrutura, adaptado de (CAD 2017).	8
Figura 2 - Dimensões BIM – adaptado de (Krona 2016)	14
Figura 3 - Níveis de Maturidade, adaptado de	15
Figura 4 - Nível de Maturidade BIM, adaptado de (BIMForum 2017)	16
Figura 5 - Níveis de Maturidade de Biljecki, adaptado de (Biljecki, Ledoux, and Stoter 2016)	17
Figura 6 - Famílias da Biblioteca Revit	18
Figura 7 - Propriedades dos elementos construtivos	19
Figura 8 - Identidade dos materiais	19
Figura 9 - Propriedades dos materiais	20
Figura 10 - Navegador de projetos do Revit	23
Figura 11 – Níveis de representação do caso de estudo	23
Figura 12 - Planos de referência	24
Figura 13 – Pilar de secção quadrada de betão (24 cm de largura e 24 cm de altura)	24
Figura 14 - Pilar de secção retangular de betão (24 cm de largura e 40 cm de altura)	25
Figura 15 - Muros de betão armado	25
Figura 16 - Template de novas famílias e pre-visualização do template escolhido	26
Figura 17 - Esquema hierárquico de famílias	27
Figura 18 - Família "Vigas de Betão"	27
Figura 19 - Laje da cobertura	28
Figura 20 – Família de Carotes cilindros 70mm	29
Figura 21 - Notação utilizada para criar o pluxograma, adotado de (Santos 2006)	32
Figura 22 - Fluxogramado para criação do Plug-In	33
Figura 23 - Estrutura de um objeto	34
Figura 24 - Código do Plug-In exemplo em C#	35
Figura 25 - Influência do Plug-In exemplo em C#	36
Figura 26 – Manipulação de valores do exercício exemplo para Dynamo	37
Figura 27 - Leitura da base de dados em Excel	38
Figura 28 - Leitura do elemento em Revit	39
Figura 29 - Blocos de decisão do primeiro Plug-In	41
Figura 30 - Diagrama de interligação entre Plug-Ins	43
Figura 31 - Notação utilizada para desenvolver o diagrama de iteração	45

Figura 32 – Digrama de interação UML	46
Figura 33 - Modelo tridimensional da estrutura em Revit	49
Figura 34 – Aparência do Dynamo Player	51
Figura 35 - Curva de calibração do esclerómetro	52

ÍNDICE DE QUADROS

Quadro 1 – Norma Diretriz Requisito BIM pelo mundo (Silva 2013).	12
Quadro 2 - Margem associada a pequeno número de resultados de ensaio	40
Quadro 3 - Coeficiente dependente do número de pares de resultados	43
Quadro 4 - Síntese dos parâmetros de avaliação.....	49
Quadro 5 - Resultado de ensaios a carotes.....	50
Quadro 6 – Resistência dos carotes e reituras correspondentes	51
Quadro 7 - Leituras do esclerómetro.....	52
Quadro 8 - Resistencia à compressão de carotes	53

ACRÓNIMOS

AEC – Arquitetura, Engenharia e Construção

AIA - American Institute of Architects

BIM - Building Information Modeling

bSa – BuildingSMART Alliance

CAD – Computer Aided Design

CGS – Government Construction Strategy

FEUP – Faculdade de Engenharia da Universidade do Porto

IAI – International Alliance for Interoperability

IFC – Industry Foundation Class

INE – Instituto Nacional de Estatística

LOD – Level of Development

PTC – Parametric Technology Corporation

PtD – Preventions through Design

URL – Uniform Resource Locator

1

INTRODUÇÃO

1.1. ÂMBITO

A crise económica que afetou Portugal nos últimos 10 anos teve um grande impacto nas áreas de produção, em especial na engenharia civil, reduzindo o interesse dos jovens por este setor. Torna-se agora possível falar sobre a sua recuperação e a forma como a manutenção e a reabilitação tem ganho uma grande importância para o crescimento da indústria da construção.

O betão é um dos materiais mais utilizados na construção em Portugal, seguindo-se a pedra e a madeira (Sampaio 2010). Sendo um material heterogéneo, a qualidade do betão é condicionada pelas propriedades das matérias constituintes e pelas condições de fabrico. Embora a maioria dos edifícios para reabilitação se encontrem no seu período de vida útil, é muitas vezes verificada a deterioração precoce do betão constituinte. Este fenómeno deve-se essencialmente a três causas: humana, ambiental e accidental. As causas humanas variam entre a fase de projeto, com uma escolha inadequada do betão ao ambiente, a fase de execução, com materiais de qualidade duvidosa e/ou uma execução deficiente, ou ainda na fase de utilização, com possíveis cargas excessivas na estrutura. As causas ambientais devem-se a ações naturais, físicas, químicas ou biológicas, que aceleram as reações dos constituintes, provocando a deterioração do betão. As causas accidentais têm origem em acidentes com sismos, cheias, incêndios ou explosões.

Neste contexto, a caracterização do estado do material, já existente ou novo em obra, é, um fator fundamental para assegurar o bom funcionamento da estrutura. A caracterização do betão pode ser efetuada através de técnicas não-destrutivas, que visam a avaliação rápida, *in situ* e sem dados para o material, ou de técnicas destrutivas que consistem na extração de amostras do betão existentes, carotes, e no ensaio até à rotura.

Os métodos de cálculo para obtenção da resistência característica do betão, segundo o instituto Português da Qualidade, são processos repetitivos que envolvem inúmeros ensaios e consequentemente resultam uma grande quantidade de informação que precisa ser tratada. A utilização de aplicações informáticas que garantem a automatização do tratamento dessa informação permite reduzir o erro e obter de forma instantânea a classe de betão existente.

As aplicações informáticas já são uma realidade no setor da construção, permitindo a modelação paramétrica, interoperabilidade entre softwares com a metodologia Building

Information Modeling (BIM) e o controlo do estado da construção durante todo o seu ciclo de vida.

1.2. MOTIVAÇÃO

A evolução da tecnologia e o crescente interesse pelas tecnologias da informação estão presentes em todos os ramos de atividades. A engenharia civil não é exceção pois, com a chegada de novos conceitos como modelação integrada em edifícios, BIM, e cidades inteligentes, a urgência de um aprofundamento do uso de tecnologias existentes, por parte dos engenheiros civis, é cada vez maior.

A rápida evolução da sociedade leva à modernização das aplicações informáticas e à flexibilidade das suas ferramentas para satisfazer as necessidades do consumidor. A metodologia BIM alcança cada vez mais seguidores e enquadra-se no crescimento atual, permitindo que estes contribuam para o constante crescimento das suas funcionalidades, através da construção de pequenos programas designados por *Plug-Ins*. Prevê-se, então, que progressivamente sejam exigidos, aos profissionais de engenharia civil, conhecimentos específicos ao nível da utilização, mas também ao nível do desenvolvimento de novas aplicações computacionais.

O interesse pela metodologia BIM, por parte da autora, e o fascínio pelas áreas de programação e otimização de processos surge da envolvimento de quase cinco anos no grupo NITEC – Núcleo de Tecnologias da Informação na Engenharia Civil, sediado no Departamento de Engenharia Civil da FEUP. Esta experiência permitiu desenvolver metodologias de algoritmia e pequenos projetos de programação, capazes de alterar a forma como um processo de ordem repetitiva pode ser encarado. Alia-se a estas competências a escolha pela especialidade de Materiais e Processos de Construção que, entre outras valências, proporcionam o desenvolvimento dos conhecimentos em materiais de construção, mais concretamente do betão.

Surge então a possibilidade de fazer convergir, neste trabalho, as áreas de interesse, potencializar o programa de modelação, *Revit*, desenvolver um *Plug-In* e automatizar a caracterização do betão existente em obra. Contribuindo para a continuação do aprendizado, mas sobretudo para a tomada de consciência da importância destas temáticas.

1.3. OBJETIVOS

Esta dissertação tem como objetivo auxiliar e automatizar a classificação do betão, abordando temáticas de armazenamento, visualização e cálculo dos valores obtidos nos ensaios. À utilização de ensaios destrutivos é associada a identificação da posição do provete retirado em obra, o armazenamento do valor obtido no ensaio e a sua utilização para obtenção da classe do betão existente. A utilização de ensaios não destrutivos, utilizando o esclerómetro de Schmidt, é iniciada pela calibração do equipamento, seguindo-se a identificação das áreas de teste e a utilização dos valores obtidos para o cálculo.

Para o cumprimento dos objetivos mencionados repartiu-se o trabalho em 4 etapas: A elaboração de elementos tridimensionais e paramétricos, recorrendo ao programa *Revit*, para que seja possível identificar a posição dos provetes e das áreas de ensaio; A construção de duas famílias, uma de provetes e outra de área de teste, para tornar identificável, na estrutura,

a posição do ensaio; A criação de um *Plug-In* capaz de efetuar os cálculos e apresentar, ao utilizador, a classe de resistência do betão; E a aplicação da metodologia em caso de estudo.

1.4. ESTRUTURA DA DISSERTAÇÃO

A presente dissertação é composta por 6 capítulos que são em seguida resumidos.

O primeiro capítulo aborda a introdução do trabalho, apresentando a âmbito de atuação, a motivação que levou à escolha deste tema, os objetivos que se pretende alcançar e a organização presente.

No segundo capítulo, intitulado Estado da arte, pretende-se retratar o panorama geral da metodologia BIM, introduzindo o conceito e explicando a sua ligação com a caracterização dos parâmetros associados aos materiais, na modelação em *Revit*,

No capítulo 3, Construção do Modelo, é explicado o processo de construção de um modelo tridimensional, através do programa de modelação *Revit*, utilizando o caso de estudo como exemplo prático.

No capítulo 4, Criação do *Plug-In*, apresenta-se o processo de criação do *Plug-In*, enumerando as possíveis alternativas para a sua execução e explicando as razões que, levaram à escolha da solução final. É ainda descrito o processo de classificação do betão, de acordo com as normas em vigor. Para ajudar à utilização do *Plug-In* foi elaborado um diagrama UML, que serve como guia de utilização.

No Caso de Estudo, correspondente ao quinto capítulo, aplica-se a metodologia a um caso de estudo real. Descrevem-se as particularidades desse caso de estudo e aplica-se o *Plug-In* desenvolvido para caraterizar o betão existente.

Por último, o capítulo 6, intitulado Conclusões e Desenvolvimentos Futuros, aborda a síntese do trabalho realizado, salientando as conclusões mais importantes e sugerindo caminhos de investigação futuros.

Segundo um estudo do INE – Instituto Nacional de Estatística – realizado em dezembro de 2017, cerca 70% da população portuguesa afirma compreender outra língua além da materna, destacando-se o inglês com cerca de 59,6% de onde 97,4% são estudantes do ensino superior (INE 2017). Conclui-se, então, que o inglês é utilizado de forma regular pelo universo académico. Uma vez que a terminologia e a bibliografia referente ao BIM é, na sua maioria, em inglês e que esta linguagem facilita a comunicação entre a comunidade científica, melhorando a compreensão do desenvolvimento de cada país, as abreviações e as imagens apresentadas nesta dissertação encontram-se maioritariamente nesse mesmo idioma.

2

ESTADO DA ARTE

2.1. BUILDING INFORMATION MODELING – BIM

2.1.1. BREVE HISTÓRIA

Embora a metodologia BIM – *Building Information Modeling* – aparente ser uma definição atual, a sua origem data de 1962 quando Douglas C. Engelbart, inspirado pelo desenvolvimento da computação, antevê um novo meio de arquitetar os edifícios (Venâncio 2015). No centro de pesquisa da Califórnia, o autor publica o “*Augmenting Human Intellect: A Conceptual Framework*” onde antecipa o conhecimento atual e fala em objetos paramétricos introduzidos em bases de dados (Engelbart 2001). Um ano depois o informático Ivan Sutherland criou um programa gráfico inovador capaz de modelar elementos baseados na geometria, o *Sketchpad*.

Em 1974, o professor Charles Eastman e a sua equipa, Tom Maver, Arto Kiviniemi e Robin Drogemuller, na consequência da criação de um novo programa de desenho aliado a uma biblioteca de elementos construtivos, relatam a noção inicial deste modelo, atribuindo a designação de BDS – *Building Description System* (Venâncio 2015).

Num artigo intitulado “*The use of computers instead of drawings in building design*” (Eastman 1975) publicado no jornal AIA – *American Institute of Architects* – em 1975, Eastman explica que a informação existente num modelo em computador permite minimizar os erros no processo de construção, aumentar a informação contida nos desenhos CAD – *Computer Aided Design* – criar bibliotecas de modelos e otimizar os métodos de execução das fases de projeto, construção e manutenção (Pelipenko e Gogina 2017, Meira 2016). Acrescenta ainda que, qualquer mudança num elemento, seria efetuada apenas uma vez para todos os elementos futuros, de modo a que “qualquer tipo de análise quantitativa poderia ser ligada diretamente à descrição (...) estimativas de custo ou quantidades de material poderiam ser facilmente geradas (...) fornecendo um único banco de dados integrados para análises visuais e quantitativas” (Venâncio 2015). Embora este conceito permitisse uma visão melhorada e evoluída da engenharia Civil, associando os seus processos a um sistema computacional, com uma base de dados ampla e garantindo um controlo virtual ao longo da vida útil da estrutura, o seu uso era limitado.

O termo BIM surge em 1992, quando os investigadores G. Nederveen e F. Tolman, inspirados por Eastman, publicaram um artigo que explica as vantagens da informação contida em modelos no processo de construção, salientando a possibilidade de intervenção de várias especialidades em diferentes fases do projeto (Nederveen e Tolman 1992).

Em 1995, é criado o formato IFC – *Industry Foundation Class* – estrutura de dados padrão que permite a transferência da informação entre *softwares* não padronizados, isto é, facilita a passagem de dados, entre programas que utilizam diferentes formatos para a definição de dados (Monteiro e Poças Martins 2013).

Em 1997, Leonid Raiz, deixa a empresa onde trabalhava, PTC – *Parametric Technology Corporation* – para conceber uma nova empresa, a *Revit Technology Corporation*. Este projeto inovador tinha como principal objetivo criar um *software* que melhorasse drasticamente a forma como os edifícios são projetados e construídos (Reece 2002).

Jerry Laiserin, analista da construção e especialista em tecnologias e estratégias de trabalho colaborativo, fez renascer, em 2002, o conceito BIM apresentando-o como uma tecnologia de vertente colaborativa e de maior eficácia face à construção tradicional. Assim é introduzido definitivamente o tema na indústria AEC – Arquitetura, Engenharia e Construção (Meira 2016). Nesse mesmo ano, a *Autodesk Inc.* reconhece o potencial da tecnologia e compra a empresa *Revit* (Bergin 2012).

Em 2006, é fundada a bSa – *BuildingSMART Alliance* – que surge da expansão da IAI – *International Alliance for Interoperability* – com o objetivo de desenvolver o formato IFC e ajudar a estabelecê-lo no mercado. Para além deste propósito, em 2007, a bSa, em parceria com o *National Institute of Building Science*, iniciou o desenvolvimento das *National BIM Standards* com o intuito de conceber um ambiente neutro entre as equipas e facilitar o desenvolvimento do nível de cadeia de valor da indústria (Hill 2007, BuildingSMART 2013).

Desde então, o BIM tem despertado o interesse da comunidade científica, originando um consequente aumento de dissertações, artigos científicos, livros e conferências. Esta atenção é vantajosa, mas, devido à acumulação de publicações, torna-se difícil avaliar, com exatidão, “a natureza exata do conhecimento descoberto, qual o impacto e contribuição e, especificamente, identificar áreas fundamentais que permanecem negligenciadas” (Hosseini 2018).

2.1.2.DEFINIÇÃO

Embora não exista uma definição consensual, entre os principais autores, sobre o BIM, uma grande maioria considera que representa um processo ou uma metodologia de trabalho, em que a ligação entre os intervenientes e as diferentes fases do projeto é fortemente melhorada (Menezes 2012, Eastman 2011, Jernigan 2007, Smith e Tardif 2009).

Esta metodologia permite adicionar, aos modelos tridimensionais representados por elementos geométricos (linhas e pontos), elevada quantidade de informação, dando significado ao objeto. Podemos destacar, por exemplo, os parâmetros geométricos, as relações com outros elementos, as características técnicas do material (peso, volume, resistências, propriedades térmicas e acústicas), o processo de instalação, os cuidados na colocação, as normas de segurança, os custos e os fabricantes (Meira 2016).

Assim, o BIM admite a integração da maioria dos intervenientes e a interoperabilidade entre diferentes plataformas. A integração é obtida pela partilha da versão mais atualizada e em tempo real, com os intervenientes do projeto, obtendo um fluxo de trabalho coletivo. A interoperabilidade permite a transferência inteligente, entre programas informáticos, de dados geométricos, semânticos e topológicos, em diferentes formatos e conteúdos, dentro de objetos paramétricos. Uma vez que cada especialidade trabalha com um *software* específico,

este aspeto permite suprimir a cópia manual das informações de um *software* para outro (Bruno 2018). A transferência do modelo detalhado para plataformas de análise estrutural, energética, sustentável e económica, de planeamento de prazos, de custos e de processos construtivos, aperfeiçoa a qualidade do trabalho e permite a obtenção de resultados meticulosos em cada etapa. Assim, BIM não é um *software*. BIM é muito mais que uma representação virtual assistida por computador do aspeto final de uma construção, é uma descrição paramétrica com informação organizada adequadamente para combinar esforços de pessoas, processos e tecnologias melhorando o modo de planear, projetar, construir, utilizar, manter e demolir uma estrutura, atualizando-se durante todo o seu ciclo de vida (Pelipenko e Gogina 2017, NIBS 2007). A Figura 1 descreve as etapas apresentadas.



Figura 1 - BIM no ciclo de vida de uma estrutura, adaptado de (CAD 2017).

2.1.3.VANTAGENS E DESVANTAGENS

Num estudo realizado por Kristen Barlish, em 2012, intitulado “*How to measure the benefits of BIM*”, a autora refere que o sucesso do BIM é relativo ao projeto e à organização. Os métodos atuais, para a avaliação dos benefícios relacionados com o sistema de informação, são insuficientes por não promoverem a visibilidade de dados comparativamente a outros projetos (Sullivan 2012). Os benefícios dependem de muitos fatores, tais como: o tamanho do projeto; a proficiência dos membros da equipa; a comunicação das equipas de trabalho; e outras condições externas organizacionais. Até ao momento, não existe uma abordagem consistente que avalie os lucros desta metodologia, ou de sistemas de informação similares, por comparação ajustada dos fatores apontados. Assim, o grau de desenvolvimento destes aspetos depende do nível de envolvimento do método, isto é, depende de empresas com

trabalhadores formados e conhecedores de BIM que, ao aplicarem o modelo na globalidade da obra, obtêm maior retorno (Construction 2014).

Contudo, o BIM apresenta inúmeras vantagens que contribuem para um produto final mais económico, eficiente e sustentável e para um processo mais produtivo e comunicativo. De forma a melhor se compreender as vantagens inerentes descreve-se, de seguida, aquelas que se considerou mais relevantes (Azhar 2011):

- Automatização – qualquer processo manual está sujeito a erro humano, mesmo quando as medições são rigorosas, não há garantia de que estejam corretas. A automatização dos processos permite o rigor na medição e na execução, bem como a atualização instantânea do modelo e a criação de plantas, vistas e cortes. Estes dados podem ser introduzidos por qualquer membro da equipa ou especialidade, explorados em processos posteriores e reutilizados noutros projetos;
- Melhor design – os desenhos desenvolvidos são uma aproximação do real, permitindo a fácil compreensão do produto final por parte do cliente e a possibilidade de reajustar ao pretendido. Os produtos finais são mais competitivos e aprimorados pois o desenho permite a simulação rápida de diferentes soluções;
- Processos mais rápidos e eficazes – a informação é inserida no objeto que a possui, podendo ser agregada e disposta de forma a facilitar a organização do utilizador. A documentação acumulada permite obter e alterar o planeamento de construção, a orçamentação e o mapa de quantidades de forma rápida e eficaz;
- Melhor qualidade de produção – todas as etapas são estudadas com rigor e detalhe, revelando menores margens de erros ou omissões e evitando as alterações de última hora frequentes na obra.
- Dados do ciclo de vida – o facto do desempenho ambiental e energético ser estimado antes da construção, por programas de avaliação automática, e anexado ao modelo tridimensional, confere uma melhor compreensão do desempenho da estrutura. Os requisitos, os materiais, os processos e as informações operacionais são arquivados e podem ser consultados, ou atualizados, ao longo do ciclo de vida. Os alertas personalizados para substituição ou manutenção de equipamentos são também implementados por facilitar a gestão temporal e orçamental do gestor do edifício. No fim de vida da estrutura, esta documentação permite a desconstrução total, fazendo um aproveitamento significativo do material e reduzindo drasticamente a libertação de gases poluentes para a atmosfera.

Apesar dos incentivos expostos, esta metodologia ainda apresenta algumas dificuldades na sua implementação. As complicações prendem-se essencialmente em quatro aspetos (Azhar 2011, Eastman 2011, Jernigan 2007):

- Investimento inicial;
- Mudança de procedimentos;
- Cooperatividade;
- Interoperabilidade.

O investimento financeiro inicial é elevado, não só pelo custo dos *softwares*, mas também pela atualização de um hardware compatível com o *software* a utilizar. Um programa que

interage com uma ou mais bibliotecas, que são constantemente atualizadas, faz cálculos e carrega cada vez mais informação e detalhe, facilmente utiliza vastos recursos computacionais como memória, disco, processador ou até placa gráfica. Quando se trabalha com modelos desenvolvidos e próximos do aspeto final deve-se ter em atenção a necessidade de utilização de um hardware capaz de corresponder às necessidades, pois quanto maior forem as funcionalidades anexadas ao modelo maior serão os recursos que ele utilizará e maior será a necessidade de um computador com características adequadas. Acrescenta-se ainda o custo da formação dos profissionais da indústria AEC que, por realizarem o seu trabalho de forma tradicional há vários anos, apresentam alguma resistência à mudança. A sofisticação dos *softwares* e a nova perspetiva inerente ao processo BIM tornam a aprendizagem lenta e complexa.

A adoção deste método representa uma mudança significativa dos procedimentos e do volume de trabalho inicial de uma empresa. O maior esforço concentra-se na criação do modelo tridimensional e na junção de detalhes, ao longo de ciclo de vida, essenciais e específicos da obra. A complexidade da implementação é tal que alguns autores sugerem diferentes estratégias de introdução do BIM nas empresas (Smith e Tardif 2009, Jernigan 2007, Eastman 2011, Arayici 2011). Apresenta-se, em seguida e de forma simplificada, o que se considerou ser as principais boas práticas:

- 1) Identificar os benefícios, tendências e riscos para a empresa – Os investidores devem reconhecer os benefícios tangíveis, as tendências da tecnologia e os riscos fundamentais para melhor procederem às mudanças necessárias quer no interior da empresa, na relação com os clientes, quer no exterior da empresa, na conexão com outras empresas. A finalidade desta etapa não é perceber como o fluxo de trabalho se pode adequar à tecnologia, mas sim como explorar a tecnologia de forma a melhorar o fluxo de trabalho;
- 2) Formar colaboradores – sensibilizar, educar e treinar os trabalhadores facilita a implementação do BIM e assegura a rentabilidade do investimento, pois são eles que assumem a responsabilidade de criar, compilar ou manter um modelo abrangente de informações para o benefício da empresa;
- 3) Fomentar uma cultura de gestão de informação – uma das características do BIM é o armazenamento de grandes quantidades de informação. Assim, não se deve esperar que alguém envolvido na obra tenha conhecimento de todos os aspetos sobre o edifício ou que perceba como a sua atividade se encaixa ou afeta o trabalho de outros. Nesta linha de pensamento, a informação, contida no modelo, tem de ser bem gerida e apresentada para que todos os intervenientes possam reconhecer que dados são relevantes para o seu trabalho;
- 4) Gerir mudanças de cultura – A adaptação cultural pode ser um dos maiores desafios desta implementação pois requer, que os trabalhadores ajustem a sua forma de trabalhar ao BIM e a todas as novas técnicas que serão implementadas. A rápida moldagem aos novos métodos proporciona um menor tempo despendido pela equipa e uma evidente economia inicial;
- 5) Atualizar a tecnologia para obter a correta migração de dados – como tem sido referido, o investimento deve acompanhar o desenvolvimento do modelo, já que este acarreta cada vez mais dados que, por sua vez devem ser suportados pelos equipamentos de leitura adequados;

- 6) Gerir expectativas – em virtude do alto investimento dirige-se, com frequência, para a criação de elevadas esperanças no rápido retorno. É fundamental capacitar toda a equipa com alguma resiliência, já que, caso algum dos aspetos anteriores não for bem-sucedido, será necessário recorrer a alterações que atrasam o retorno pretendido;
- 7) Medir o progresso até ao objetivo estratégico – fazer medições do caminho já percorrido, do que falta percorrer, de como lá chegar e que melhorias são possíveis efetuar é o trabalho constante dos gestores, com o intuito de fazer chegar o processo de implementação ao objetivo delineado na etapa inicial;
- 8) Aceitar novos paradigmas de negócio – a tecnologia BIM é, por si só, um novo paradigma de negócio, logo a indústria AEC não deve estagnar só porque a metodologia já foi implementada, mas deve sim, procurar atualizar-se constantemente.

A cooperatividade é outro aspeto que se pode tornar numa desvantagem. Note-se que, para tirar o melhor proveito possível do BIM, é necessário que todos os intervenientes da obra tenham a metodologia implementada (Meira 2016). Assim a partilha de documentos flui facilmente e a correção de incompatibilidades é imediata, ou seja, basta que um dos participantes não possua o mesmo nível de implementação BIM para que todos os outros vejam o seu trabalho comprometido ou não usufruam de todas as capacidades do modelo.

A interoperabilidade é a capacidade de troca de informação entre programas de forma a dinamizar, automatizar e cessar a repetição de dados introduzidos pelas diferentes equipas. Uma vez que todas as equipas têm um trabalho específico e distinto, cada interveniente tem a possibilidade de escolher o *software* que melhor se adequa ao seu trabalho (Ferreira 2015). A comunicação entre plataformas distintas é executada de duas formas: 1) quando os dois sistemas são compatíveis, isto é, utilizam a mesmo formato proprietário para definição de dados, a informação é transferida automaticamente, sem perdas de dados; 2) quando ambos os sistemas usam diferentes formatos proprietários, a troca de dados não acontece automaticamente, é necessário converter a informação para um terceiro sistema comum, geralmente o IFC (Sullivan 2012). Este último processo apresenta erros e omissões consideráveis que têm levado a uma redução do investimento na metodologia BIM.

Com o intuito de regularizar os processos de modelação e acordar a responsabilidade entre os participantes da indústria AEC, algumas organizações desenvolveram normas ou diretrizes para o uso do BIM. O Quadro 1 representa uma lista não exaustiva da adoção do BIM em todo o mundo, permitindo ter uma visão clara da forma como esta metodologia tem proliferado. Observa-se que não só as grandes potências económicas, como os Estados Unidos, ajudam a desenvolver normas e diretrizes, apesar destes possuírem muitas organizações e universidades “são os países com uma indústria de construção mais pequena que lideram a implementação do BIM” (Silva 2013).

Quadro 1 – Norma | Diretriz | Requisito BIM pelo mundo (Silva 2013).

País	Organização	Nome da Norma Diretriz Requisito	Data de Publicação
Austrália	NATSPEC	<i>NATSPEC National BIM Guide</i>	19 set. 2011
		<i>NATSPEC BIM Object/Element Matrix</i>	
Dinamarca	Erhvervsstyrelsen (National Agency for Enterprise and Construction)	<i>Det Digitale Byggeri (Digital Construction)</i>	1 jan. 2007

Finlândia	BuildingSMART	<i>Common BIM Requirement 2012 (COBIM)</i>	27 mar. 2012
Reino Unido	AEC	<i>AEC (UK) BIM Protocols</i>	7 set. 2012
Noruega	Statsbygg	<i>Statsbygg Building Information Modeling Manual</i>	24 nov. 2011
Singapura	Building and Construction Authority	<i>Singapore BIM Guide</i>	15 mai. 2012
	CORENET e-submission system (ESS)	<i>CORENET BIM e-submission Guidelines</i>	25 jan. 2010
Estados Unidos da América	National Institute of Building Science (NIBS) – BuildingSMART alliance (bSa)	<i>National BIM Standard (NBIMS)</i>	4 mai. 2012
	American Institute of Architects (AIA) Contract Documents	<i>E202-2008 BIM Protocol Exhibit</i>	2008
	New York City Department of Design + Construction	<i>BIM Guidelines</i>	1 jul. 2012
	United States Department of Veterans Affairs (VA)	<i>The VA BIM Guide</i>	2 abr. 2010
	Indiana University Architect's Office and Engineering Services	<i>IU BIM Guidelines & Standards for Architects, Engineers, and Contractors</i>	2 jul. 2012
	BuildLACCD (Los Angeles Community College District)	<i>BIM Design-Bid-Build Standards BIM Design-Build Standards</i>	29 jun. 2011
		<i>LACCD BIM Standard</i>	2 jan. 2010
	United States General Services Administration	<i>National 3D-4D Building Information Modeling Program</i>	15 mai. 2007

Destaca-se a *buildingSMART alliance* por gerir a *Nacional BIM Standard* – Estados Unidos. Esta norma resulta da compilação de vários requerimentos e regras que vão sendo lançados por diferentes entidades. Por a bSa fazer parte da *BuildingSMART* internacional, a norma propagou-se rapidamente sendo utilizada como base para a criação de novas normas e conta, atualmente, com cerca de trinta mil downloads por 131 países. A versão mais atualizada desta norma, V3 (terceira versão), surge a 22 de julho de 2015 pelo consenso da comissão e cobre todo o ciclo de vida dos edifícios, desde o planeamento e projeto à construção e operações (Standard 2015). Quando uma nova versão é lançada para o mercado surgem críticas e recomendações por parte dos utilizadores que são filtradas e servem de base de trabalho para a criação da versão seguinte. Deste modo, a empresa garante que as versões são cada vez mais satisfatórias e que a evolução segue as necessidades dos consumidores.

2.1.1.DIMENSÕES DOS MODELOS

Por o BIM promover o detalhe e a informação concedida a um modelo tridimensional, o nível mais baixo aqui apresentado é o 3D. Torna-se evidente que o 1D e o 2D são dimensões de desenho, no entanto não se enquadram na metodologia por serem apenas pontos e linhas, que produzem uma forma, e não um modelo orientado para o objeto e para a finalidade enquanto entidade em obra. Assim, apresenta-se as seguintes dimensões:

BIM 3D – Modelo paramétrico – Representa um modelo BIM quando, ao desenho 3D, é adicionado particularidades sobre o objeto, que promovam o enriquecimento do desenho e permitam alcançar um modelo inteligente.

BIM 4D – Planeamento – Ao modelo BIM 3D acresce a variável tempo que possibilita detalhar o faseamento construtivo das tarefas (Jongeling e Olofsson 2007). Esta dimensão admite que o desenho tridimensional simplifique a leitura da preparação do trabalho, dividindo-o por níveis, especialidades ou categorias de forma a facilitar a interpretação dos intervenientes e interessados. A mais valia reside na facilidade de controlar ou alterar a cronologia global da obra ajustando e conferindo os prazos de pequenas tarefas (Dawood 2003).

BIM 5D – Orçamento – Ao modelo anterior, BIM 4D, é acrescentado um dos aspetos cruciais de uma obra, o custo. Esta dimensão conjuga automaticamente o valor financeiro final (custos associados à compra, aplicação e mão de obra) de cada material ou elemento com as quantidades e com o faseamento construtivo, tornando acessível a simulação de diferentes soluções antes ou durante a obra e a escolha dos materiais que conduzem a um maior controlo de custos. A perceção de qual fase é mais dispendiosa é também consentida, dando destaque aos componentes que mais contribuem para o custo final e qual a sua localização na linha temporal (Qiqi 2016, Jiang 2017).

A partir deste nível as opiniões divergem, há autores que agrupam a dimensão 6D à gestão de instalações e outros que a associam à sustentabilidade. Os primeiros alegam que a sustentabilidade deve ser tida em conta em todas as fases do projeto e em todas as dimensões, sem destacar uma para esse fim (Bakowski 2017, Krona 2016). Por outro lado, os segundos autores, explicam o peso que a sustentabilidade tem, sobretudo nos dias atuais, associada à proteção da natureza, ao destacar esta visão numa dimensão individual (Nical e Wodynski 2016). Contudo, em concordância com os primeiros autores, foi criada uma sétima dimensão de modo a assegurar a gestão e manutenção da estrutura durante o seu tempo de vida.

BIM 6D – Sustentabilidade – Ao modelo 4D adiciona-se a avaliação e certificação de mérito ambiental. Esta dimensão foca-se no consumo de energia: na melhoria da eficiência energética, no aumento de consumo de energia produzida a partir de fontes renováveis e na redução das emissões de gases com efeito de estufa. Atualmente este controlo é assegurado pela União Europeia que apresenta um quadro de objetivos energéticos e climáticos perspetivados para os próximos anos (2020, 2030 e 2050) (Comissão Europeia 2014).

BIM 7D – Gestão e manutenção da estrutura – Esta dimensão tem em consideração a fase mais longa da vida de um edifício, a utilização. Por este motivo torna-se extremamente útil aproveitar o modelo que vai sendo atualizado durante a projeção e a construção da obra. Neste patamar é possível obter-se informações dos fabricantes e fornecedores, dos tipos de materiais utilizados, características, referências e garantias (Venâncio 2015). Ao tipo de construção utilizada pode-se adicionar alarmes e lembretes de possíveis manutenções programadas e estimar-se o estado de conservação do edifício.

Seguindo a lógica aqui apresentada já se fala de uma nova fase ligada à prevenção através do design, *PtD – Preventions through Design* – Esta oitava dimensão garante uma maior segurança dos trabalhadores da construção civil, diminuindo os casos de risco e consequentemente a taxa de acidentes. Embora este potencial ainda não tenha sido explorado, estudos provam que a capacidade de influenciar a segurança diminui com o avanço da obra, devendo-se apostar numa avaliação precoce dos riscos envolvidos (Kamardeen 2010). A imagem seguinte esquematiza as dimensões até agora exibidas.



Figura 2 - Dimensões BIM – adaptado de (Krona 2016)

2.1.4. NÍVEIS DOS MODELOS BIM

O nível de modelação interfere, francamente, com o agravamento de erros e omissões que vão ocorrendo em cascata durante as várias fases do projeto. Ao carecer de detalhe, os modelos menos desenvolvidos apresentam menor precisão na estimativa de custos, na planificação global e no nível de sustentabilidade. Destacam-se os níveis de maturidade e os níveis de desenvolvimento por serem os que maior impacto apresentam na interoperabilidade. Estes permitem superar as lacunas e garantir que todas as partes se encontram a trabalhar no mesmo patamar de modelação.

2.1.4.1. NÍVEIS DE MATURIDADE

Este conceito foi criado no Reino Unido com a intenção de caracterizar o grau de implementação da metodologia nas empresas AEC. Os quatro níveis, ilustrados na Figura 3, distinguem as empresas quanto à qualidade e eficácia do modelo e à partilha de informação entre colaboradores (Bouška 2016):

No nível 0, o grau mais primitivo, a empresa trabalha com modelos CAD, apresenta apenas modelos 2D, e a produção e troca de informação entre as especialidades é realizada maioritariamente em papel. Logo não é aplicada a metodologia BIM.

No nível 1, a empresa projeta as obras em modelos CAD para visualização 3D. A partilha entre as especialidades não é colaborativa e a informação é gerida individualmente, embora possa haver utilização de *softwares* de planeamento e custos. Por se utilizar algumas ferramentas do BIM, mas não haver conexão entre as equipas, este nível também é conhecido como “*lonely BIM*”.

No nível 2, a empresa utiliza as metodologias BIM mais básicas, apenas para troca de modelos com caracterização dos objetos. Este é o grau mínimo exigido pelo Reino Unido, desde 2016. O CGS – Government Construction Strategy – estabeleceu este relatório com o objetivo de ver reduzidos, entre 15 a 20%, o custo das obras públicas (Cabinet Office 2011).

No nível 3, a empresa aplica o BIM em todas as fases de projeto, tirando melhor partido desta tecnologia. A informação é gerada no modelo e partilhada, em tempo real, com todas as especialidades.

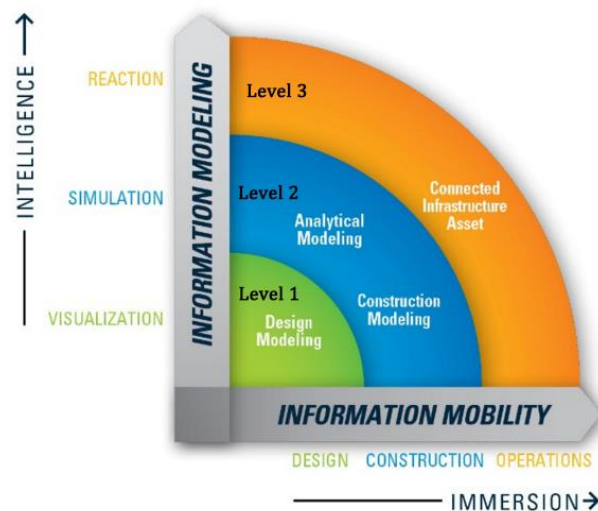


Figura 3 - Níveis de Maturidade, adaptado de (Manziona 2016).

2.1.4.2. NÍVEIS DE DESENVOLVIMENTO

O nível de desenvolvimento BIM vem do inglês LOD – *Level of Development* – e pretende oferecer, aos profissionais da indústria AEC, ferramentas que proporcionem uma articulação acessível do grau de clareza e de confiabilidade do modelo BIM, no decorrer dos diferentes processos de construção. Com esta noção é possível identificar os objetos que possuem informação suficientemente desenvolvida para a partilha eficiente entre equipas, pois modelos com diferentes níveis de desenvolvimento obtêm resultados distintos em análises de cálculo automático, uma vez que ostentam diferentes representações gráficas.

Os seis níveis apresentados pela AIA, no novo ciclo de documentos de prática digital lançados em junho de 2013, com a “finalidade de documentar os protocolos e procedimentos acordados que irão reger o desenvolvimento, a transmissão, o uso e a troca de elementos num projeto” (AIA 2013), são explicados de seguida.

No LOD 100 os elementos não são representações gráficas, mas sim símbolos ou interpretações geométricas aproximadas que transmitem uma ideia simplificada do que se pretende construir.

No LOD 200 os objetos do modelo são uma representação gráfica elementar que retêm noções aproximadas de volume, forma, quantidade, localização ou orientação. Neste momento os desenhos são espaços reservados para possível aperfeiçoamento futuro.

Os elementos no LOD 300 já não são formas aproximados. Neste nível de desenvolvimento já revelam uma configuração específica onde os dados anexados: quantidade, tamanho, forma, localização, orientação e outros, são exatos e as medições podem ser exercidas no modelo.

O LOD 350 é uma melhoria do nível anterior, onde a interface com outros sistemas de construção é possível e a configuração geométrica é ainda mais precisa pois, os elementos próximos são coordenados e conectados. Por exemplo, numa viga de betão armado apresentaria, não só a geometria exata, mas também o posicionamento da armadura no seu

interior. As medições podem ser exercidas no modelo e os cálculos automáticos já são representativos.

No LOD 400 o elemento é uma representação fiel do almejado, expondo, para além dos dados já mencionados, orientações promissoras para os processos de fabricação, montagem e instalação.

Por último, no LOD 500 a peça desenhada é uma réplica exata do real, com toda a minúcia gráfica e física. Este LOD é apenas uma verificação do campo, ou seja, não corresponde a uma evolução para um nível mais alto, pois comporta as variações na fase de gestão e manutenção da estrutura. Por estes motivos, a sua representação na figura 4 é, em tudo, semelhante ao LOD 400.

Desde 2011, o *BIMForum* compilou especialistas BIM nas áreas de engenharia e arquitetura no sentido de produzir ilustrações gráficas dos níveis de desenvolvimento baseados no delineado pela AIA. O objetivo deste grupo é padronizar o uso consistente do esquema LOD, aumentando a sua utilidade e a colaboração entre trabalhadores desta indústria. Do seu mais recente trabalho, de novembro de 2017 (*BIMForum* 2017), foi adaptada a Figura 4.

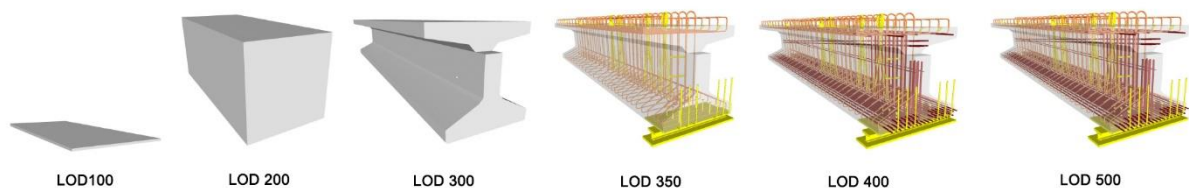


Figura 4 -Níveis de desenvolvimento BIM, adaptado de (*BIMForum* 2017)

Contudo, outros autores, não concordam com esta visão simplificada e acrescentam mais níveis de desenvolvimento. Filip Biljecki, no artigo “An improved LOD specification for 3D building models” (Biljecki 2016) aponta dezasseis LODs que acredita serem diferentes de um ponto de vista geométrico e que produzem diferentes resultados quando transferidos para

programas de cálculo automático. Os dezasseis níveis de desenvolvimento de Biljecki podem ser consultados na Figura 5.

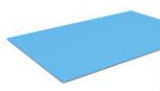
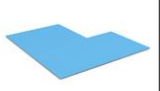














	LOD x.0	LOD x.1	LOD x.2	LOD x.3
LOD 0				
	LOD 0.0	LOD 0.1	LOD 0.2	LOD 0.3
LOD 100				
	LOD 100.0	LOD 100.1	LOD 100.2	LOD 100.3
LOD 200				
	LOD 200.0	LOD 200.1	LOD 200.2	LOD 200.3
LOD 300				
	LOD 300.0	LOD 300.1	LOD 300.2	LOD 300.3

Figura 5 - Níveis de desenvolvimento de Biljecki, adaptado de (Biljecki 2016)

2.2. MATERIAIS COM INFORMAÇÃO BIM

A representação de objetos com informação é facilmente apresentada por *softwares* de desenho adaptados à tecnologia BIM. Para melhor se compreender o processo de construção de um elemento inteligente escolheu-se o *Revit*. Este *software* inclui, numa única interface, as funcionalidades de quatro especialidades: arquitetura, estrutura, construção e MEA - mecânica, eletricidade e água. Foi especialmente construído para modelação em BIM, capacitando os utilizadores de ferramentas para criar projetos desde do seu conceito inicial à conceção final. Com uma abordagem colaborativa, cooperativa e baseada em interação de parâmetros, este programa é acessível e gratuito para qualquer estudante.

O *Revit* dispõe de uma ampla biblioteca, de elementos mais utilizados, já modelados e com algumas características associadas. Esses elementos encontram-se organizadas por famílias, como se observa na Figura 6, que apresentam funcionalidades e parâmetros semelhantes, como por exemplo: fundações, pilares, vigas, paredes, portas e janelas. Um objeto pode ser criado de raiz, copiado ou editado. Ao copiar um elemento cria-se um novo objeto, exatamente igual ao primeiro, mas onde se pode alterar as suas propriedades sem danificar a base de dados. Editar altera as especificidades já existentes nesse objeto e criar um novo elemento, ou uma nova família, implica recriar a sua geometria e vincular

especificidades e parâmetros para que seja possível empregar o elemento em qualquer projeto. Apesar da maioria das bibliotecas disponíveis serem editáveis, o *Revit* possui uma biblioteca bloqueada, protegida contra modificação ou exclusão do seu conteúdo. Desta forma, utilizadores que pretendam editar qualquer elemento dessa biblioteca podem duplicá-lo e editar a sua cópia. Cada usuário pode conceber as suas próprias bibliotecas, sendo capaz de bloqueá-las caso pretenda partilhar com outros membros de equipa sem perder a informação. A exportação de bibliotecas para outros *softwares* é sempre exequível, embora nem todos os programas possuam uma leitura dos dados tão ampla e bem definida como o *Revit*. Então, poderá haver perdas de informação.

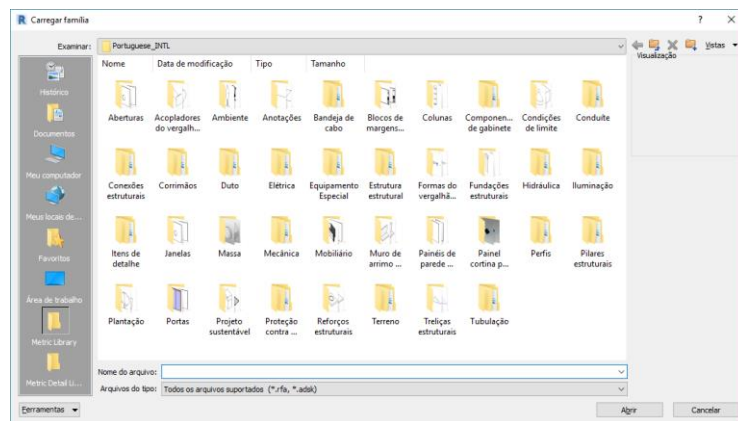


Figura 6 - Famílias da Biblioteca Revit

As propriedades de um objeto ostentam as restrições, a estruturação, as cotas, os dados de identidade, a fase e o material. As restrições referem-se a limitações impostas pelo utilizador ao elemento de modo a torná-lo interativo, isto é, ao limitar a posição de um elemento em relação a outro, obriga-se a que, caso haja alteração no segundo elemento, o primeiro seja automaticamente adaptado. Estas restrições podem ser relativas à posição da base ou do topo, ao deslocamento superior, à distância da extensão superior, entre outros. Ao fixar, por exemplo, a posição inferior de um pilar a uma planta de piso estamos a obrigá-lo a manter-se sempre nessa posição. Se, por algum motivo, a cota da planta for modificada, o objeto é ajustado espontaneamente a essa nova cota. As propriedades estruturais dizem respeito à função do elemento. Caso seja um elemento estrutural, é possível editar o recobrimento dos varões de aço em cada face do elemento. As cotas orientam o utilizador para o comprimento do objeto, para a área que este ocupa e para o volume associado. Os dados de identidade remetem para aspetos de coletividade, como a marca do elemento, comentários do utilizador ou uma imagem relevante para uma melhor compreensão do modelado. Aspetos relacionados com a fase dirigem o utilizador para a criação de diferentes fases do ciclo de vida da estrutura como demolição, nova construção e manutenção. São exemplos os dois elementos representados na Figura 7.

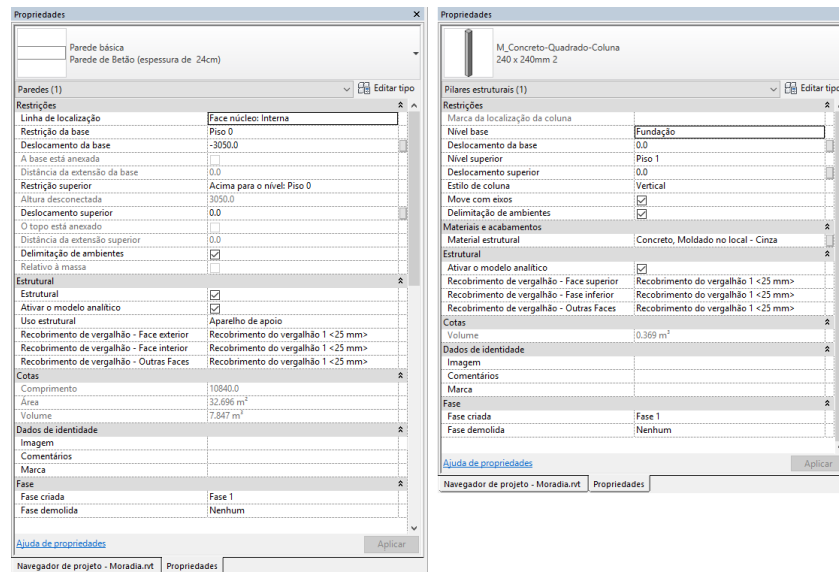


Figura 7 - Propriedades dos elementos construtivos

Cada modelo pode conter mais do que um material, com características bem distintas, que são facilmente introduzidas recorrendo ao navegador de materiais. As propriedades dos materiais são organizadas em identidade e em recursos.

A identidade, descrita na Figura 8, apresenta-nos o nome do material, as informações descritivas referentes à classe do material como madeira, vidro, betão, cerâmica e metal, as informações sobre o produto como o fabricante, o modelo, o custo e uma ligação URL – *Uniform Resource Locator* – direcionando o utilizador para o site de encomenda, e as informações de anotação do *Revit* que se referem à marca e às notas-chave que atuam como anotações importantes ou indicações para o futuro, são exemplos: “Divisória à prova de poeira temporária”, “Relva existente a permanecer”, “Remover construção existente”, “Janela e estrutura a serem removidas”, “Pavimento rígido” e “Base plana para absorção de choque”.

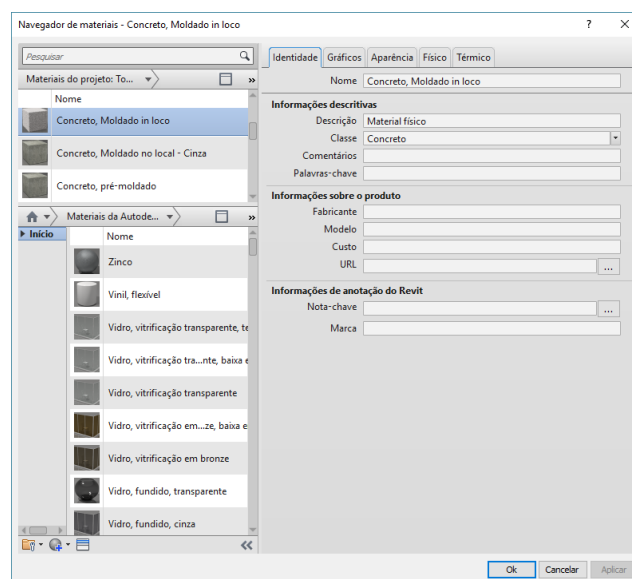


Figura 8 - Identidade dos materiais

Os recursos são grupos de propriedades que comportam determinadas características ou componentes do objeto. O *Revit* é o único *software* da *Autodesk* que possui quatro tipos de recursos, como indicado na Figura 9: os gráficos, a aparência, as propriedades físicas e as propriedades térmicas. Os dois primeiros controlam a aparência exibida em diferentes tipos de vistas e cortes. Este *software* considera o módulo de elasticidade, densidade, resistência à compressão e resistência à tração como propriedades físicas, que são utilizadas para análise estrutural. Do mesmo modo, considera a condutividade térmica, a permeabilidade, a porosidade e a resistência elétrica propriedades térmicas, que são fundamentais para a análise de energia.

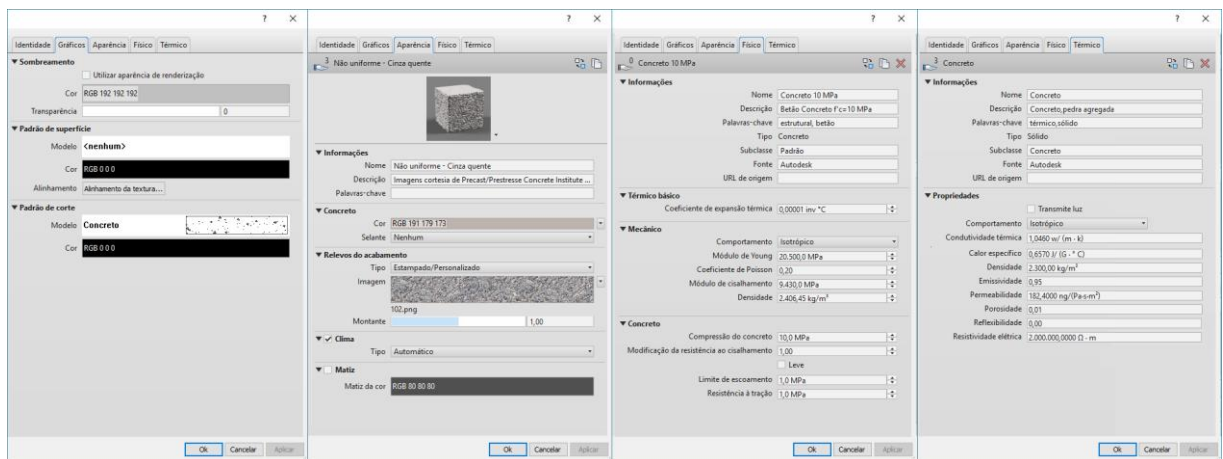


Figura 9 - Propriedades dos materiais

A *Autodesk* dispõe de uma unidade de *Knowledge Network* online que viabiliza a aprendizagem de qualquer tema relacionado com este programa.

3

CONSTRUÇÃO DO MODELO

Neste capítulo pretende descrever-se a construção do modelo tridimensional e paramétrico, utilizado a geometria bem definida do caso de estudo. Este modelo possibilitará a criação de qualquer outra iniciativa com base nesta.

Ao iniciar a criação de um novo modelo, o *Revit* sugere, automaticamente, a escolha do tipo de especialidade que irá basear o novo projeto: construção, arquitetura, estrutura ou MEA. Para profissionais do setor da construção, o programa permite entender melhor os meios, os métodos e os materiais a utilizar e até analisar a viabilidade da construção antes de a iniciar. Engenheiros de estruturas dispõem de ferramentas específicas para a conceção, criação, análise de segurança e cooperação com outras componentes de construção. Arquitetos podem desenhar as suas ideias de projeto, otimizar e visualizar o seu desempenho até à construção. Engenheiros mecânicos, eletrotécnicos e hidráulicos projetam sistemas distintos, mas com a mesma filosofia. O *Revit* dispõe das ferramentas necessária e permite, com maior precisão, aliar as componentes arquitetónicas e de especialidade, nunca perdendo a informação e a coordenação inerentes aos modelos BIM. Pretende-se que a escolha seja o mais adequada possível ao objetivo do trabalho, uma vez que cada especialidade exibe ferramentas específicas para esse propósito. Para este trabalho optou-se por um modelo de construção, por ser a especialidade mais próxima da área em que este documento se encontra inserido.

Ao abrir uma especialidade surge, na lateral esquerda, o Navegador de Projeto, retratado na Figura 10, onde se encontram as vistas, as tabelas, as famílias e os grupos. A pasta Vistas engloba as plantas do projeto, que representam os pisos existentes e/ou outros cortes que o utilizador considere necessários; as elevações, que permitem navegar pelo projeto alterando a perspetiva do visualizador, apresentando numa vista 2D as quatro orientações possíveis: norte, sul, este e oeste; e a vista 3D. A pasta Tabelas/Quantidades agrupa de forma automática as componentes específicas do projeto em curso. Analisando, a título de exemplo, a tabela “Quantidades de equipamento especializado” verifica-se que inclui para cada elemento da lista, o código de montagem, a descrição de montagem, a contagem de elementos existentes do mesmo tipo, a descrição da unidade, o tipo de elementos, o fabricante, o modelo e, entre outros, o endereço URL. A pasta Famílias apresenta uma lista de famílias e elementos que foram carregadas automaticamente no projeto. À medida que o utilizador for carregando outras famílias, a lista será atualizada.

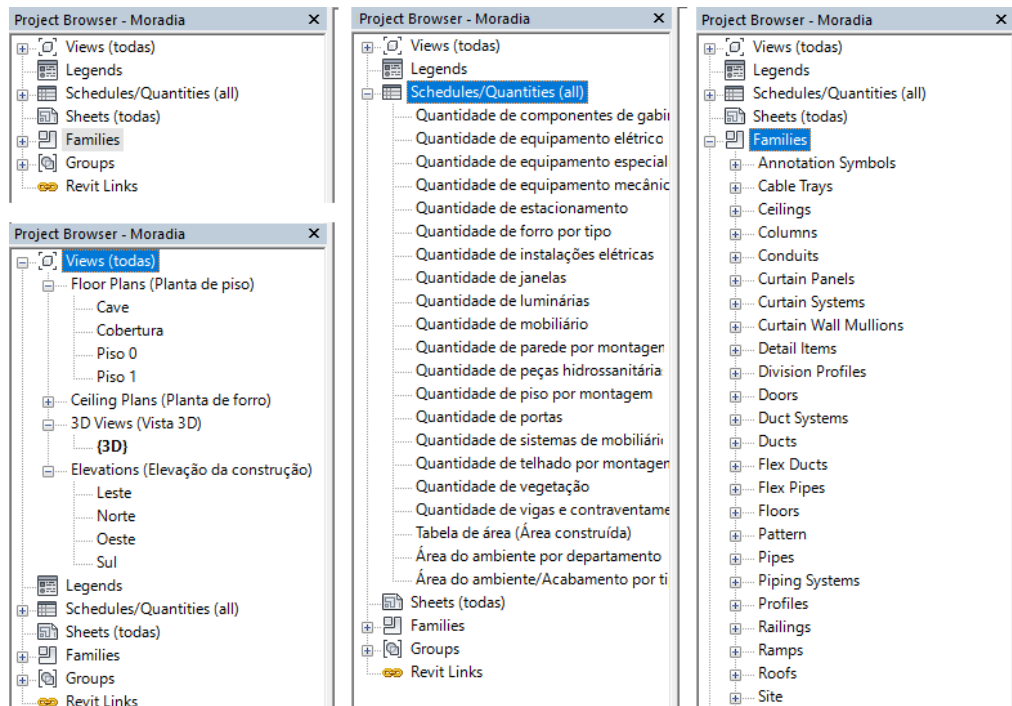


Figura 10 - Navegador de projetos do Revit

3.1. MODELAÇÃO DE PILARES E PAREDES

Inicia-se a modelação por abrir a pasta Elevações que contem as vistas: Norte, Sul, Este e Oeste, escolhe-se uma delas e verifica-se que são automaticamente exibidos os níveis pré-existentes. Cada especialidade apresenta níveis iniciais distintos.

Níveis são planos tridimensionais visíveis apenas nas elevações que intersejam, encontram-se afastados por cotas e influenciam a posição dos elementos construtivos. Alguns elementos construtivos são limitados pelos níveis, como pilares e paredes, outros são contidos em níveis, pisos e vigas. O caso de estudo refere-se a uma moradia de três pisos, logo foram inseridos quatro níveis: Cave, Piso 0, Piso 1 e Cobertura, como evidencia a Figura 11. É recomendado etiquetar e cotar devidamente estes elementos para que haja clareza nos desenhos e sejam facilmente reconhecidos durante o desenvolvimento do projeto. Agora, estes níveis são as plantas e são acessíveis através da pasta Plantas de Piso contida em Vistas.



Figura 11 – Níveis de representação do caso de estudo

O *Revit* dispõe de linhas e planos de referência que auxiliam a criação do modelo. Esta ferramenta é crucial para garantir o correto posicionamento de cada família e simplificar a leitura em planta. A utilização de um plano vertical permite visualizar uma linha, que compõe o plano, em cada piso, apoiando a representação entre níveis. A moradia apresenta apenas elementos estruturais: muros, pilares, vigas e lajes, todos com diferentes geometrias. Alguns já existem na biblioteca pré-instalada da *Autodesk*, outros serão alterados e, no caso das vigas, será necessário construir uma nova família. Na Figura 12 são evidenciados os planos de referência auxiliares à construção do caso em estudo.

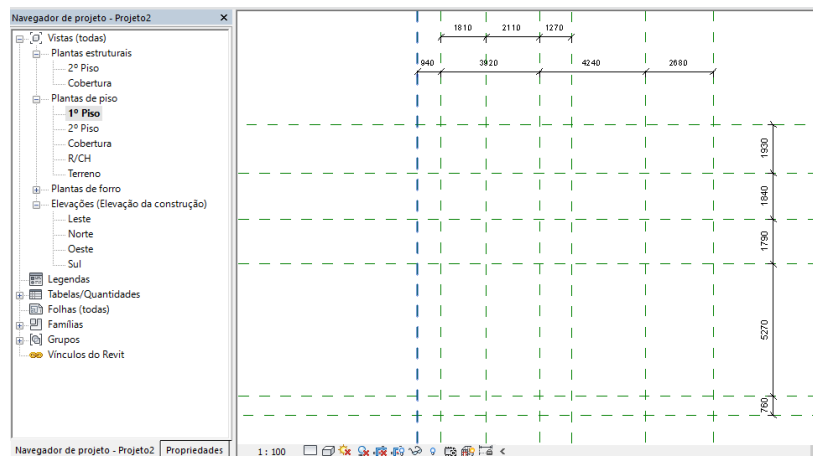


Figura 12 - Planos de referência

No cruzamento de alguns dos planos de referência são inseridos os pilares limitados pelos níveis correspondentes. A biblioteca já possui uma família de pilares de betão, contudo as dimensões não coincidem com o especificado no caso de estudo. Será necessário duplicar um dos elementos carregados e editar as medidas para obter os dois tipos da Figura 13 e Figura 14.

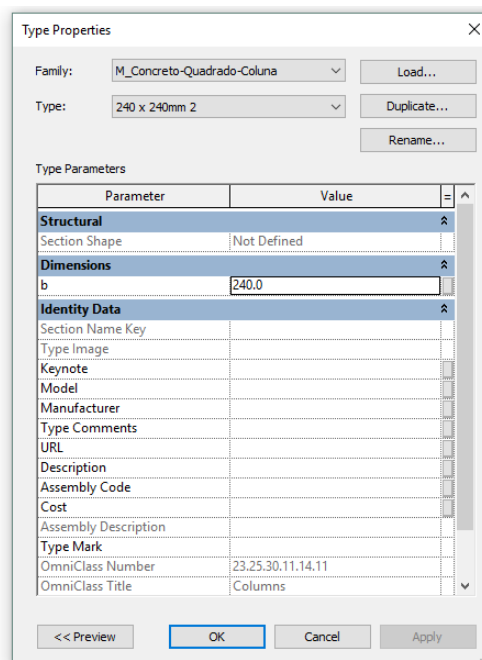


Figura 13 – Pilar de secção quadrada de betão

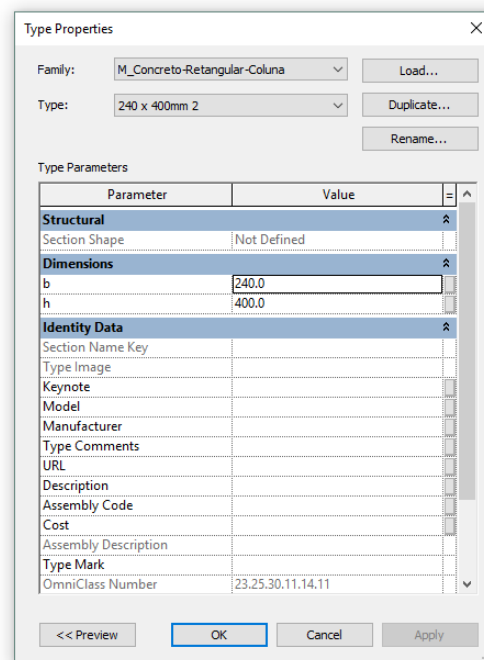


Figura 14 - Pilar de secção retangular de betão

A moradia ostenta muros de betão armado, localizados na cave e que oferecem apoio a duas lajes do piso 1. No painel “Estrutura”, do programa, é possível carregar paredes estruturais para o projeto aberto, já que o Revit conserva uma extensa biblioteca de paredes exteriores e interiores com um só material, simples, ou complexa, com mais de dois materiais. Os muros de betão, representados pela Figura 15, foram modelados recorrendo a uma parede genérica da biblioteca. Posteriormente foram editados e foi ajustada a sua espessura.

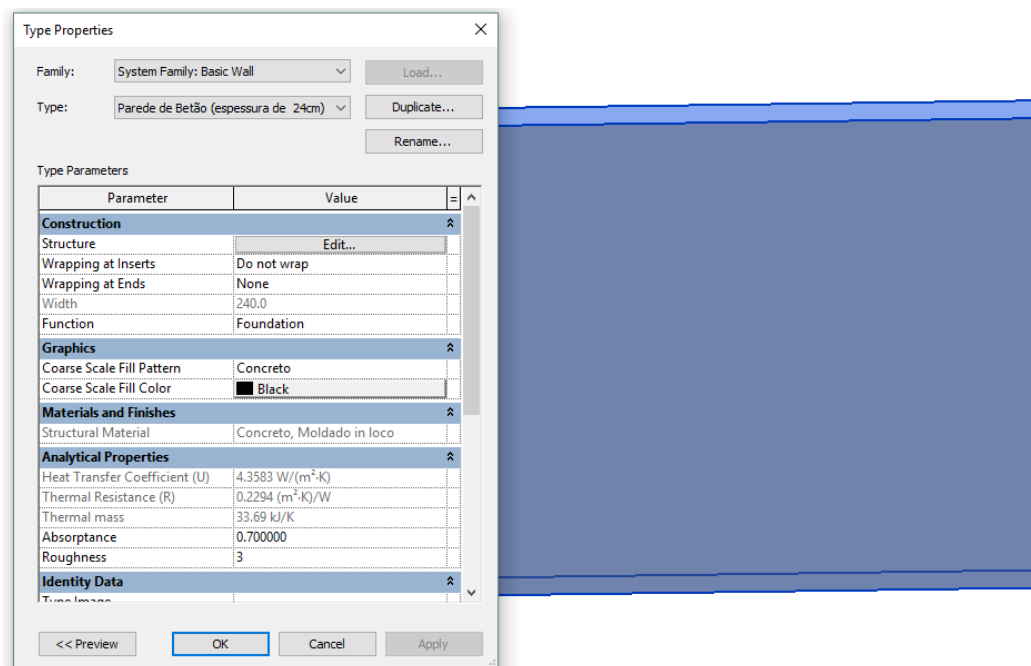


Figura 15 - Muros de betão armado

3.2. MODELAÇÃO DE VIGAS

O caso de estudo recorre a vigas de betão armado, algumas embebidas e com geometrias nitidamente distintas. O *Revit* está preparado para a construção de vigas, mas apenas apresenta, na biblioteca, vigas de perfil metálico em I pois a maioria dos edifícios do Reino Unido são assim construídas. Então, será necessário criar uma nova instância de vigas para o efeito. Com o intuito de incentivar a criação de bibliotecas, por parte dos utilizadores, a *Autodesk* disponibiliza uma biblioteca de *templates* específicos para cada objetivo e, se por qualquer motivo, não existir o *template* exato para a intenção do usuário é possível escolher *templates* genéricos, sem qualquer vínculo às famílias pré-existentes.

No menu Iniciar, no canto superior esquerdo, recorrendo ao painel Novo e selecionando Família é revelada a pasta de *templates* pré-instalados pela *Autodesk*. Optou-se por uma modelo de “reforço estrutural métrico com base em linha” que apresenta três linhas de referência pré-desenhadas, duas centradas, que pretendem ser os eixos de referência do elemento a criar e uma descentrada, com um parâmetro de comprimento, como é possível visualizar na Figura 16.

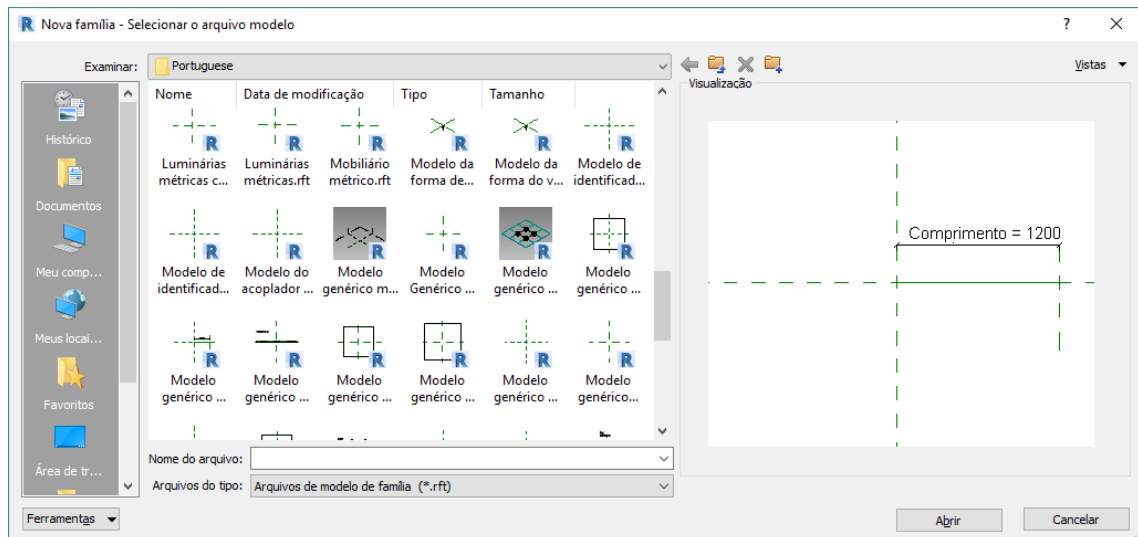


Figura 16 - Template de novas famílias e pre-visualização do template escolhido

No menu Criar, é apresentada uma série de ferramentas próprias para desenho. O utilizador deve escolher a que melhor se adequa à concessão esperada. Neste caso foi utilizada a ferramenta de Extrusão, foi desenhada a secção retangular da viga e a peça foi centrada nos eixos de referência. Para a construção pretendida as linhas de referência foram suficientes, mas para elementos mais complexos é possível criar mais planos de referência auxiliares. Segue-se a identificação dos parâmetros da família. Esta ferramenta, um pouco complexa, é a base da tecnologia BIM por consentir o relacionamento de diferentes características do elemento que, antes, não possuíam qualquer ligação. Existem dois tipos de parâmetros distintos: os de Tipo e os de Instância. Os primeiros são relativos a um elemento da família, como os pilares retangulares de betão. Os segundos correspondem a cada cópia desse elemento, como o pilar retangular de betão e de medidas (altura x base). Modificações cometidas nos Tipos afetam todas as cópias desse tipo, inseridas no projeto. Alterações

associadas às Instâncias alteram meramente aquela cópia. O esquema da Figura 17 permite clarificar o exposto.

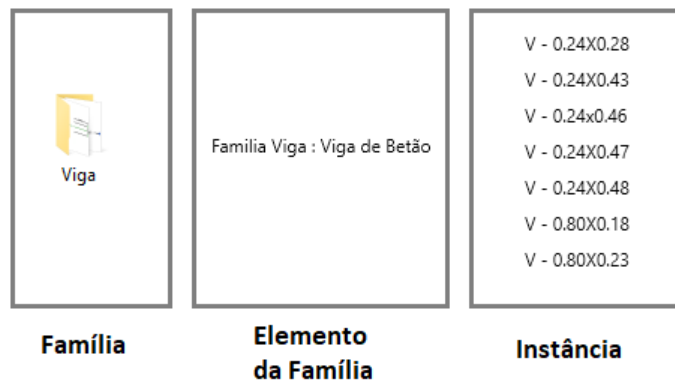


Figura 17 - Esquema hierárquico de famílias

Além do nome, os parâmetros estão agrupados em: Disciplinas, como elétrica, estrutural e energia, por exemplo; Tipos de Parâmetros que abrangem áreas, volumes, texto, material, imagem, entre outros; e Parâmetros de Grupo como cotas, dados ou restrições. Nenhum deles é meramente exclusivo, pelo que cada parâmetro deve dispor de uma categoria em cada grupo. Cabe ao utilizador perceber que conjunto de categorias, existentes em cada grupo, levará a um melhor produto final. Pois, por exemplo, para criar parâmetros de visualização é fundamental escolher, no grupo Tipo de Parâmetros, a categoria Sim/Não ou para criar parâmetros de dimensões é necessário escolher a categoria Linear, no mesmo grupo. Parâmetro de dimensões viabilizam a ligações entre diferentes medidas do elemento, isto é, quando se pretende que a seção da viga se mantenha retangular, base x altura, por exemplo, ao permitir a alteração de uma das medidas a outra deve ajustar-se a essa alteração, aumentando ou diminuindo, mas mantendo a proporcionalidade imposta na sua criação. Para o elemento da família "Viga de Betão" criou-se os dois parâmetros esquematizados na Figura 18: altura e largura, para além do comprimento, já pré-definido. Foram criadas outras instâncias do mesmo tipo de família, alterando apenas os valores geométricos, de acordo com o caso de estudo, para que não seja necessário modificar constantemente a família no decorrer da modelação. Em seguida, a Família é introduzida no projeto e cada viga é colocada na respetiva posição.

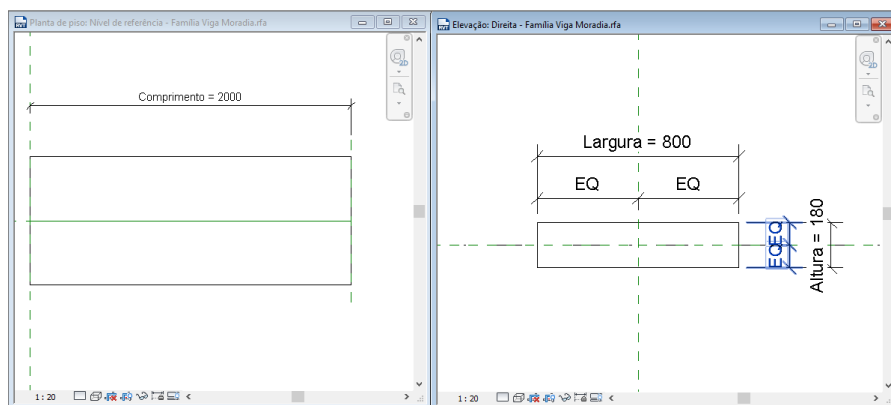


Figura 18 - Família "Vigas de Betão"

3.3. MODELAÇÃO DE LAJES

A modelação das lajes em *Revit* pode ser efetuada de três formas distintas: como laje estrutural, no menu de Estruturas, como chão ou como teto, no menu de Arquitetura. Como o modelo tridimensional é apenas representativo, optou-se por desenhar cada laje com um método diferente. Assim, tem-se conhecimento do processo de modelação de cada opção.

A laje do Piso 0 foi modelada utilizando a ferramenta *Slab* do menu Estruturas. Para tal, escolhe-se a planta correspondente e com o retângulo do separador *Draw*, desenha-se a laje com as dimensões desejadas.

A laje do Piso 1 foi desenhada segundo a ferramenta *Floor* do menu Arquitetura. Do mesmo modo, escolhe-se a planta do piso correspondente, Piso 1, e com a ferramenta de desenho dá-se a forma pretendida.

A laje da Cobertura foi configurada tendo por base a ferramenta *Ceiling* do menu Arquitetura. Desta vez escolhe-se o piso inferior, Piso 1, para se desenhar o teto, mas a modelação da laje é semelhante às restantes opções.

Antes ou depois das lajes serem modeladas é necessário introduzir a sua espessura, duplicando e editando uma instância da família Lajes de Betão. A Figura 19 exibe a laje modelada para a cobertura.

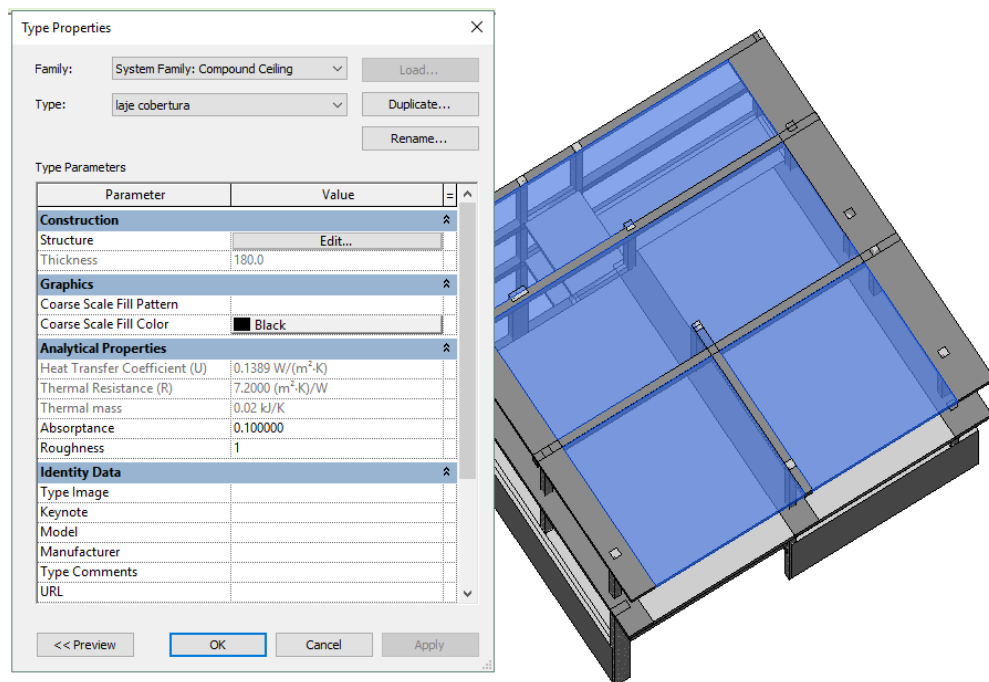


Figura 19 - Laje da cobertura

3.4. MODELAÇÃO DE CAROTES E PONTOS DE LEITURA DO ESCLERÓMETRO

Segundo o processo de criação de famílias, utilizado para a criação do elemento da família “Viga de Betão”, criou-se a família “Carotes cilíndricas 70mm”. Trata-se de carotes cilíndricos com 70 mm de diâmetro, de acordo com as carotes retiradas no caso de estudo. Assim, depois da escolha do modelo de “reforço estrutural métrico com base em linha”, existente na pasta *Templates* do *Revit*, desenhou-se o provete da Figura 20.

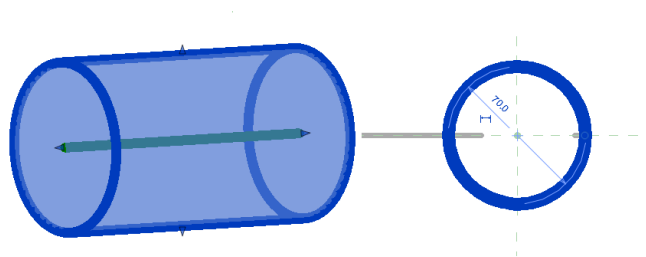


Figura 20 – Família de Carotes cilíndricas 70mm

De igual modo procedeu-se à criação da área de leitura do esclerómetro. Foram desenhados 9 pequenos volumes de 50 mm de aresta, que correspondem às 9 leituras mínimas impostas pela norma. A esta nova família deu-se o nome de “Leituras (9)”.

As novas famílias foram carregadas para o modelo da moradia e colocadas nas respetivas zonas ensaiadas no caso de estudo. A família “Carotes cilíndricas 70mm” foi representada a azul e a família “Leituras (9)” foi representada a vermelho.

4

CRIAÇÃO DO *PLUG-IN*

As aplicações informáticas modernas tendem a ser cada vez mais flexíveis, mas não estão preparadas para satisfazer todas as necessidades dos seus utilizadores. Com vista a dotá-los de mais alguma versatilidade criaram-se os *Plug-Ins*.

Um *Plug-In* é uma aplicação que, a um programa informático base, acrescenta funcionalidades ou novas características não existentes. Frequentemente sobre forma de um comando que automatiza uma tarefa ou personaliza um comportamento. Embora a sua origem date de 1970, a palavra *Plug-In* é, hoje em dia, muito explorada por programadores, empresas e utilizadores particulares (Autodesk 2018c). Confirma-se este facto com a sua extensa disponibilidade no mercado atual. A maioria dos *softwares* ostenta uma lista alargada de *Plug-Ins*, pagos ou gratuitos, que permite adaptar o programa às necessidades de cada um. Serve de exemplo o *Plug-In* mais instalado em *web browseres*, como *Google Chrome*, *Mozilla Firefox*, *Microsoft Edge* ou *Safari*: O *Codec de Vídeo*. Quando se pretende abrir um vídeo ou um áudio on-line, a versão original dos navegadores não está habilitada à leitura destes formatos. Então é sugerida, automaticamente, a instalação do *Plug-In* apropriado (Richard 2010). Outro exemplo comum são os inúmeros *Plug-Ins* disponíveis para o *WordPress*. Este gestor de conteúdos para web, voltado para criação e gestão de sites, permite a instalação de pequenas aplicações que, entre outras funcionalidades, podem servir para modificar a sua estética. Os *Plug-Ins* são executados pelo programa principal e requerem uma linguagem de programação compatível com as duas partes. Mesmo assim, por vezes surgem conflitos que interferem com o bom funcionamento do *software*. É facilmente confundido “extensão” com “*Plug-In*”, embora representem funcionalidades diferentes. Um *Plug-In* adiciona recursos que permitem personalizar o programa base e pode ser concebido por qualquer utilizador. Uma extensão é um programa concebido para criar ou modificar algumas funcionalidades, como a nova versão de um programa, e é frequentemente projetado pelos criadores do *software* (Richard 2010).

O *Revit* é uma plataforma programada em código fechado. Isto significa que o utilizador não tem acesso ao seu código de criação, logo, não pode reajustar, por este meio, as funcionalidades do *software* às suas necessidades. Os *Plug-Ins* vêm preencher esta lacuna e contribuir para o constante avanço de um serviço ou de uma tecnologia. Neste caso pretende-se capacitar a plataforma *Revit*, introduzida na metodologia BIM, a ajudar o utilizador a identificar a qualidade de betão existente em obra. Imagine-se que, no decorrer de uma obra, o engenheiro pretende verificar uma das seguintes situações: qual a classe de betão

que acaba de chegar à obra ou qual a classe de betão existente. No primeiro caso o utilizador irá produzir provetes e ensaiá-los. No segundo caso poderá utilizar o esclerómetro de Schmidt para testar a resistência superficial do betão ou retirar carotes para análise em laboratório. Imagine-se ainda que este utilizador dispõe de um computador com o software instalado e o *Plug-In* aqui desenvolvido. Neste caso, é possível, após a obtenção dos resultados de ensaio e da sua introdução no *Revit*, obter imediatamente a classe de resistência do betão correspondente.

O *Revit* apresenta dois métodos preferenciais, compatíveis com o programa, para a criação do *Plug-In*: a conceção de um código em C# (Autodesk 2018c) ou a elaboração de um esquema em *Dynamo BIM* (Autodesk 2016). O C# é uma linguagem de programação, com vários anos de existência. O *Dynamo* é uma interface de programação recente e visual.

Para se decidir qual o melhor método para o desenvolvimento do *Plug-In*, exploraram-se as duas opções e, posteriormente, optou-se por uma delas. A análise destas duas linguagens e a justificação da preferência são descritas neste capítulo. Embora uma das opções incida na programação em C#, por a autora já possuir alguns conhecimentos nesta linguagem, o *Revit* também é adaptável com outras linguagens de programação compatíveis com o *.NET framework*, como VB.NET ou F# (Autodesk 2018c). O *.NET framework* é uma tecnologia desenvolvida pela *Microsoft*, para o *Windows*, que fornece, entre outros serviços, extensas bibliotecas de código reutilizável. A grande mais valia desta tecnologia é a capacidade de converter rotinas ou códigos, escritos numa linguagem de programação, noutras linguagens, admitindo a escolha da linguagem por parte programador (Microsoft 2018).

4.1. CRIAÇÃO DO ALGORITMO

O desenvolvimento de qualquer *Plug-In*, passa por diferentes tarefas: o planeamento, a análise, a programação, a implementação e a manutenção. Para cada fase foram estudados métodos e desenvolvidas técnicas específicas para que melhor e mais rapidamente se chegue ao resultado final esperado (Santos 2006). O planeamento e a análise são abordados neste subcapítulo, a programação será discutida no subcapítulo 4.5, no capítulo 5 será apresentada a implementação do *Plug-In* e a manutenção será explanada no capítulo 6: Conclusões e desenvolvimentos futuros.

O planeamento consiste na descrição pormenorizada, faseada e refinada das etapas a que o programa está sujeito, definindo um conjunto de regras que, juntamente com variáveis e processos de decisão, levam à melhor organização possível. O método mais comum para o processo de planeamento é a criação de um *algoritmo*. Esta estratégia consiste na manipulação de dados acedendo, organizando e alterando o modo como os dados são apresentados. Um *algoritmo* é um conjunto de comandos finitos, bem definidos, que descrevem os passos lógicos necessários à realização de uma tarefa ou resolução de um problema. É recorrente fazer-se o paralelismo entre uma receita de culinária e um *algoritmo* pois o primeiro engloba ingredientes e instruções de preparação e o segundo contém dados e instruções de manipulação. Então um *algoritmo* assemelha-se a uma receita que descomplica a construção de um programa ou um *Plug-In*. Dos dois métodos mais comuns, fluxograma e pseudo-código, escolheu-se o fluxograma por auxiliar a descrição, de forma gráfica, e ainda, ostentar uma representação simbólica que espelha os diferentes tipos de instruções básicas: sequência, decisão e repetição. A primeira baseia-se numa sequência de instruções que permitem a leitura e a escrita de dados, juntamente com o cálculo e a atribuição

de valores. O segundo fundamenta-se na decisão/seleção de uma alternativa, que desencadeiam um conjunto de ações, depois da avaliação lógica da condição introduzida. A instrução de repetição apoia-se no ato de repetir um conjunto de ações, que, em cada ciclo é testado o valor lógico da condição introduzida, até não ser necessária outra iteração (Santos 2006). Para planeamento do *Plug-In* foram utilizadas as notações ilustradas da Figura 21.

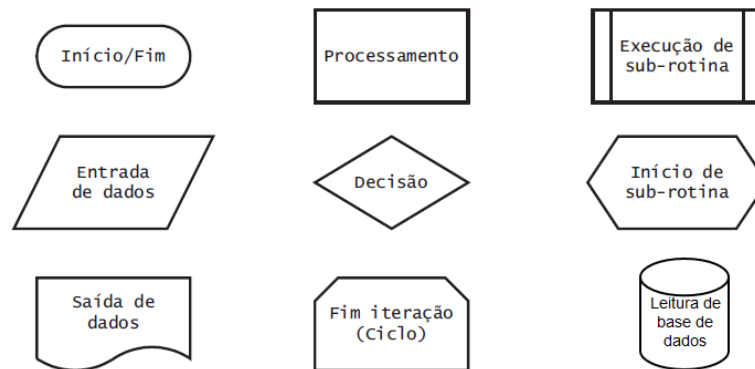


Figura 21 - Notação utilizada para criar o fluxograma, adotado de (Santos 2006)

Da notação apresentada surgiu o esquema da Figura 22. O *algoritmo* está dividido em duas partes: a seleção do elemento e a escolha do ensaio a utilizar.

Na primeira parte o *Plug-In* deve ser capaz de identificar o elemento selecionado pelo utilizador: Viga, Pilar, Muro ou Laje. Deve procurar na base de dados os materiais compatíveis com esse elemento e apresentar a primeira lista de materiais possíveis.

Na segunda parte o utilizador deve escolher que ensaio pretende utilizar, de acordo com os dados inseridos no modelo. De acordo como o caso de estudo, optou-se por programar o *Plug-In* para a análise de dois tipos de dados: Resistências à compressão de carotes, de acordo com a norma EN 1239-3, e a resistência superficial obtida pelo esclerómetro de Schmidt.

É crucial, depois do planeamento, fazer-se uma análise do seu melhoramento, ponderando o que poderia ser acrescentado ou até o que está presente e não é fundamental. Esta observação foi efetuada e permitiu obter uma meta mais alargada do funcionamento do *Plug-In*. Uma vez que a norma EN 1239-3 também engloba o ensaio de compressão uniaxial de provetes, opou-se por alargar a área de atuação do *Plug-In* para englobar estes casos. E, antes da utilização do esclerómetro é necessário proceder à sua calibração para o betão existente em obra, assim este processo também consta dos procedimentos disponibilizados pelo *Plug-In*.

A Figura 22 apresenta o fluxograma descritivo do que se pretende desenvolver com o *Plug-In*.

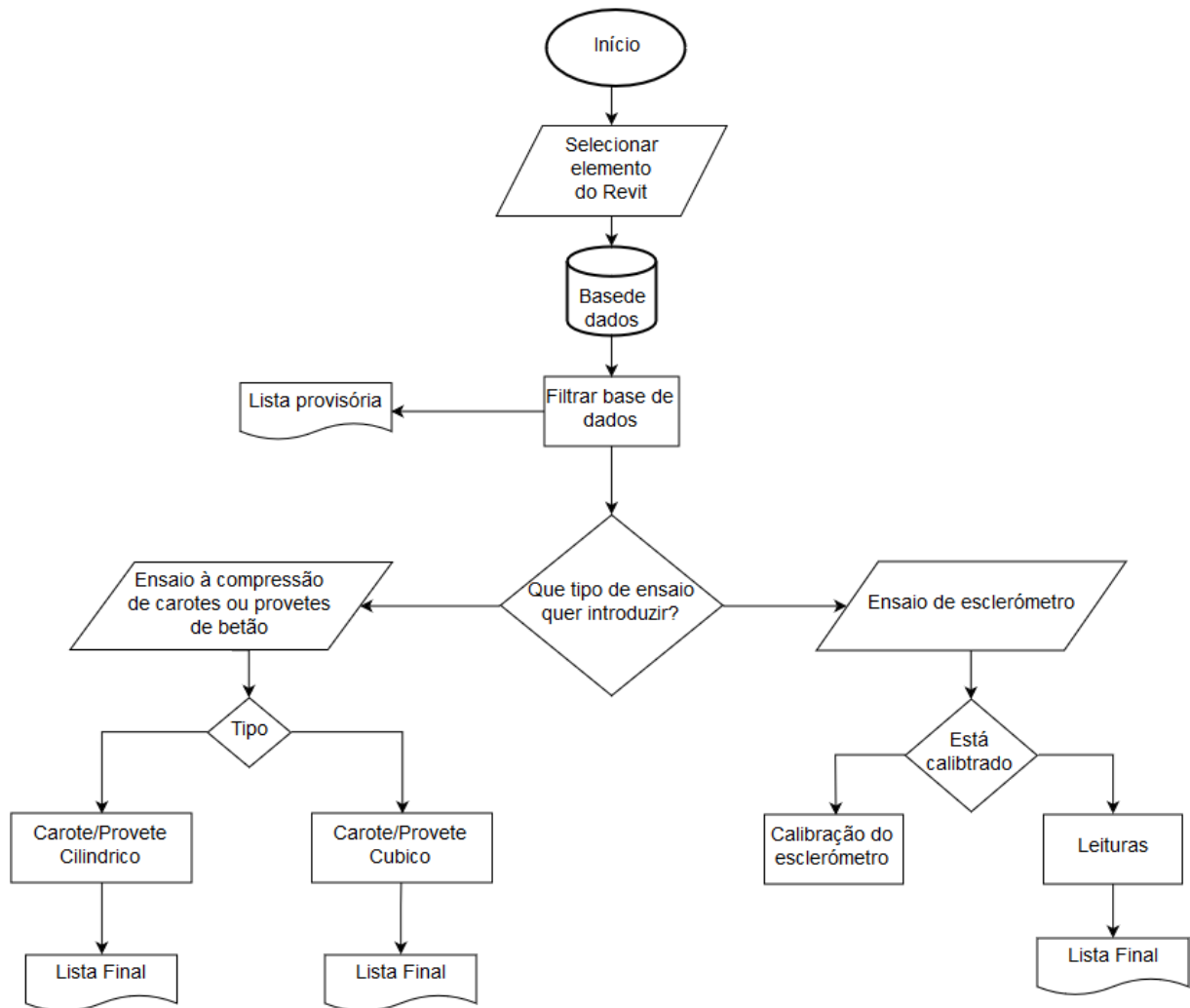


Figura 22 - Fluxogramado para criação do Plug-In

4.2. INICIAÇÃO AO C#

O C# é uma linguagem de programação simples e orientada a objetos. Isto significa que, quando se pretende executar ações simples de uso corrente, é possível criar um objeto que o faça automaticamente. Quando for necessário voltar a executar essa ação, basta chamar o objeto. Este processo poupa tempo ao programador e evita a reprogramação e a repetição de funções amplamente utilizadas. Cada objeto tem estados e um ou mais métodos. Os estados são os dados e as informações nele contidas; os métodos são as funções que ele executa (Microsoft 2017). Contudo, os objetos são frequentemente confundidos com classes. Para melhor se compreender a diferença entre os dois conceitos, vamos assemelhá-los ao processo de construção de uma casa: A classe é a planta da casa, é um esquema abstrato que define “o que” se vai construir e “qual” a posição desses elementos, determina “o que” será modelado e “quais” serão os métodos que o objeto pode conter; um objeto é a casa em si, é algo concreto e temporário. Nele os elementos estão de facto presentes. O objeto guarda memória, possuindo valores para os estados. É possível compreender melhor a estrutura de um objeto com esquema da Figura 23.

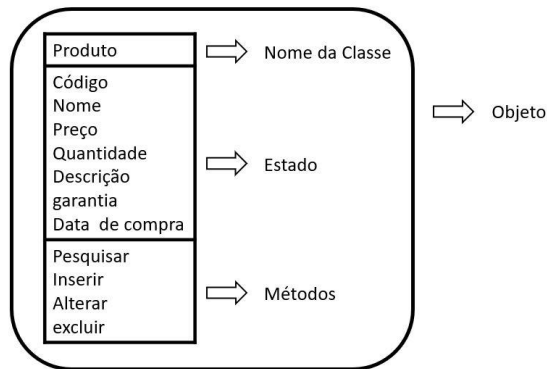


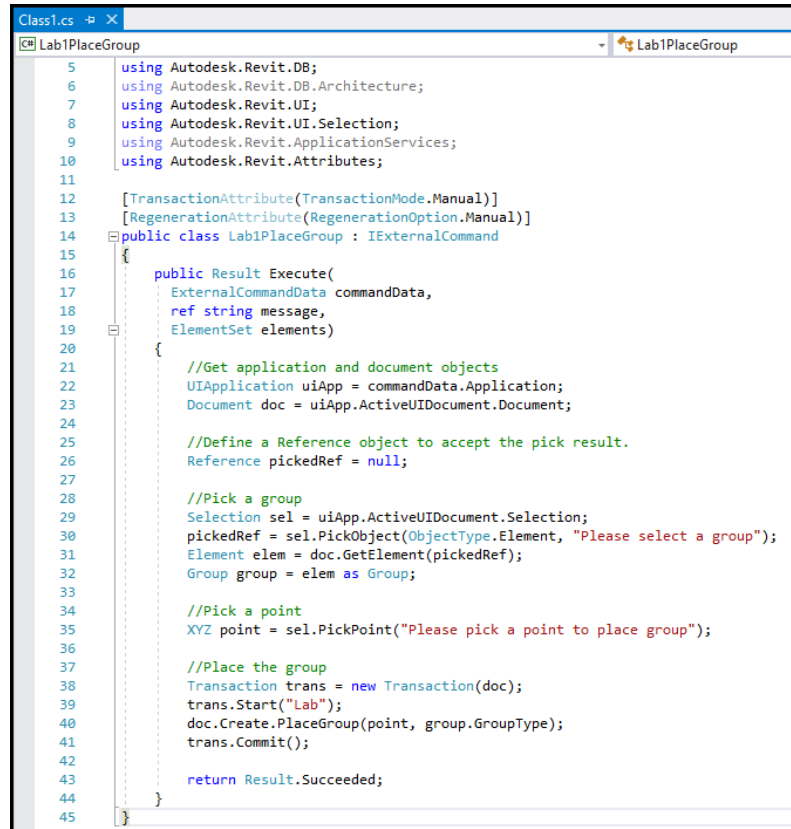
Figura 23 - Estrutura de um objeto

Por ser uma linguagem orientada a objetos, o C# engloba conceitos como encapsulamento, herança e polimorfismo. O encapsulamento é o processo que garante a consistência do tipo de informação retirada do objeto. Quando se cria um objeto deve-se pensar na sua utilização final, mas também nas alterações que esse fim pode sofrer. Assim, deve-se ocultar o acesso ao estado interno do objeto e implementar outros membros, como conteúdos e propriedades, para que haja encapsulamento do estado (Mayo 2016c). A herança é um dos principais conceitos da programação orientada a objetos pois permite a reutilização de código já existente. Este tipo de programação permite conceber uma hierarquia, de classes associadas, que transferem a informação integrada. À associação entre estas classes é vulgarmente chamada de *ParentClass* e *ChildClass*. Assim, à criação da classe principal, *ParentClass*, é possível incluir outras classes com funcionalidades distintas, que variam da primeira, *ChildClass*, e que podem solicitar os métodos da classe que lhe deu origem (Mayo 2016a). O polimorfismo é a capacidade de juntar, a um *array*, objetos de tipos distintos e invocar, posteriormente, os seus métodos. Um *array* é um conjunto de informação acumulada em repartições distintas, como um vetor ou uma matriz, que recebe valores ou objetos (Mayo 2016b). Assim, esta linguagem de programação é favorável à atribuição de grupos de objetos, que não necessitam de ser do mesmo tipo, mas que, se estiverem relacionados por herança, é viável adicioná-los à matriz como o tipo herdado, compartilhando o mesmo nome do método.

Para melhor se compreender a ligação entre o C# e o Revit, iniciou-se a criação de um *Plug-In* exemplo através da plataforma “*My first Plug-In*” disponibilizado pela Autodesk (Autodesk 2018c). Esta ferramenta pretende familiarizar os utilizadores de Revit, independentemente das competências de programação, com o Revit .NET API – *Application Programming Interface*. A API de um software é a forma de comunicação entre programador e o programa, que permite que se escrevam instruções e que elas sejam executadas posteriormente. A maioria das APIs possuem um conjunto de bibliotecas, criadas pelo fornecedor, que permite interagir de forma mais direta com o software. Porém, o tipo de programa a desenvolver depende, não só do quão desenvolvidas estão as bibliotecas, mas também de como a API foi projetada (Autodesk 2018c).

Na elaboração de um modelo arquitetónico é recorrente haver disposições construtivas semelhantes a que chamamos: layouts comuns. Nesse caso, o *Plug-In* que pretendemos criar tem como objetivo copiar o *layout* decorativo de um quarto de hotel e implementá-lo noutro

quarto. Após a instalação do *Visual Studio*, e dos pacotes existentes para programação em C#, criou-se uma biblioteca de classes onde se adicionou as referências da API do *Revit* e, por fim, introduziu-se o código explanado na Figura 24.



```

Class1.cs
Lab1PlaceGroup
5  using Autodesk.Revit.DB;
6  using Autodesk.Revit.DB.Architecture;
7  using Autodesk.Revit.UI;
8  using Autodesk.Revit.UI.Selection;
9  using Autodesk.Revit.ApplicationServices;
10 using Autodesk.Revit.Attributes;
11
12 [TransactionAttribute(TransactionMode.Manual)]
13 [RegenerationAttribute(RegenerationOption.Manual)]
14 public class Lab1PlaceGroup : IExternalCommand
15 {
16     public Result Execute(
17         ExternalCommandData commandData,
18         ref string message,
19         ElementSet elements)
20     {
21         //Get application and document objects
22         UIApplication uiApp = commandData.Application;
23         Document doc = uiApp.ActiveUIDocument.Document;
24
25         //Define a Reference object to accept the pick result.
26         Reference pickedRef = null;
27
28         //Pick a group
29         Selection sel = uiApp.ActiveUIDocument.Selection;
30         pickedRef = sel.PickObject(ObjectType.Element, "Please select a group");
31         Element elem = doc.GetElement(pickedRef);
32         Group group = elem as Group;
33
34         //Pick a point
35         XYZ point = sel.PickPoint("Please pick a point to place group");
36
37         //Place the group
38         Transaction trans = new Transaction(doc);
39         trans.Start("Lab");
40         doc.Create.PlaceGroup(point, group.GroupType);
41         trans.Commit();
42
43         return Result.Succeeded;
44     }
45 }

```

Figura 24 - Código do Plug-In exemplo em C#

O código é composto por seis pequenos grupos fáceis de identificar pela indentação utilizada no código. O primeiro começa por introduzir as bibliotecas necessárias ao correto funcionamento do *Plug-In* e definir como será executada a transferência de *inputs* que, neste caso, são processados manualmente através do clique do botão do rato. O segundo inicializa os objetos utilizados na aplicação e no documento para que, mais tarde, seja possível aceder aos seus estados e métodos. O terceiro grupo procede à definição do objeto através da função “pickedRef”, identificando-o inicialmente como nulo, pois irá receber a informação contida no quarto de hotel selecionado pelo utilizador. No quarto trecho utiliza-se funcionalidades do *Revit* para escolher o grupo a copiar e guardar essa informação no objeto criado anteriormente. De seguida, na quinta secção, o utilizador escolhe a posição onde pretende introduzir a informação contida no objeto. Por último, no sexto grupo, é efetivamente colada a informação.

Seguiu-se a implementação do *Plug-In* no *Revit*, este processo requer que o ficheiro de leitura em C# seja guardado numa pasta pré-instalada especificamente, pela *Autodesk*, para o efeito. Posto isto, chegou a altura de experimentar o *Plug-In*, para tal basta abrir o separador “suplementos” de *Revit* e procurar nas “ferramentas externas” a lista de *Plug-Ins* já criados.

Utilizando qualquer modelo é possível verificar a eficácia da aplicação, neste caso foi utilizado um modelo de um hotel disponibilizado para teste. Na Figura 25 apresenta-se a evolução instantânea do modelo, recorrendo a esta ferramenta.

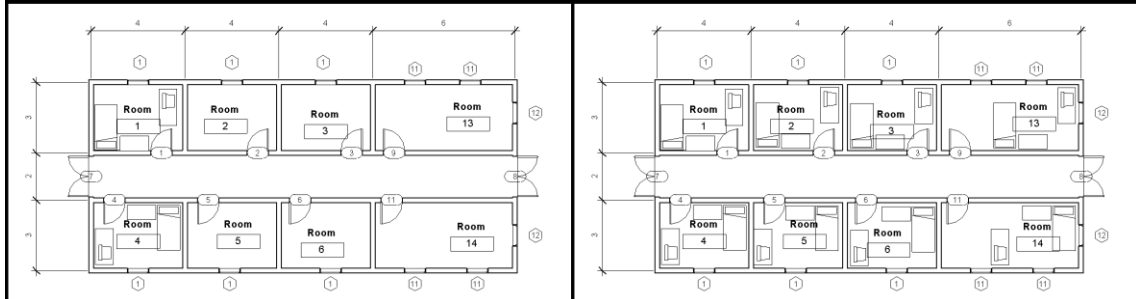


Figura 25 - Influência do Plug-In exemplo em C#

4.3. INICIAÇÃO AO DYNAMO

O *Dynamo* é uma interface de programação visual, a sua primeira versão foi lançada em 2016, pela Autodesk como um complemento, de BIM, para o *Revit* (Autodesk 2018b). É também uma plataforma *open-source*, o que permite a qualquer indivíduo, descarregar o *software* gratuitamente, contudo as versões mais recentes do *Revit*, 2017 a 2019, já incluem a instalação automática do *Dynamo*. Esta linguagem de programação permite, a utilizadores não especializados, desenvolver os seus próprios *Plug-Ins*, numa interface simples e intuitiva. Alia-se o facto de proporcionar a manipulação de dados, esculpir novas geometrias, explorar novas ferramentas, automatizar processos e personalizar o fluxo de trabalho, que concedem, ao programa, um comportamento técnico e focado na obtenção de conhecimentos específicos da indústria AEC (Sanhudo e Poças Martins 2016).

Criar uma aplicação em *Dynamo* é como criar uma sequência de tarefas, a cada tarefa corresponde um comando que irá levar, passo a passo, ao produto final. A conexão entre o *Revit* e o *Dynamo* é realizada por outra ferramenta BIM, o *Dyno*, este *software* serve de interface à programação criada pelo *Dynamo* no *Revit*. Para o *software* operar corretamente não é necessária grande informação pois a maioria dos inputs são informações obtidas diretamente do modelo em *Revit* (Sanhudo e Poças Martins 2016). Apesar da simplicidade de programação podem existir processos sem correspondência com as funções existentes, então, para resolver este aspeto o *Dynamo* permite a introdução de código em *Python*. Esta é outra linguagem de programação focada na produtividade e na clareza, é também caracterizada pelo seu baixo uso de caracteres especiais, pouca utilização de palavras de compilação e pelo uso lato de blocos (*Python* 2018).

Há semelhança do ocorrido na linguagem C#, para melhor se compreender o conceito inserido nesta ferramenta, desenvolveu-se um pequeno exercício de utilização do *Dynamo*. O objetivo é criar um cilindro recorrendo apenas à programação. Iniciou-se a procura das funções, no motor de busca do *Dynamo* ou no *Dynamo Dictionary online* (Autodesk 2017) que poderiam gerar o sólido, selecionou-se a função que recolhe os inputs: raio, ponto de início e ponto de fim: “*Cylinder.byPointsRadius*”. A esta função foram adicionadas as funções: “*Point.Origin*” como o ponto de início, “*Point.ByCoordinates*”, como coordenadas alteráveis de fim e as

funções “*Number Slider*” como reguladores de input. Cada função ou bloco inclui nós de saída e/ou de entrada, chamados *Nodes*, que são ligados através dos fios, *Wires*, que, por sua vez, conectam as funções e originam o fluxo de dados. Agora é fácil manipular os valores do raio, da posição final do X ou do Z a fim de se obter diferentes combinações, que geram diferentes cilindros. Duas das manipulações efetuadas são representadas pela Figura 26.

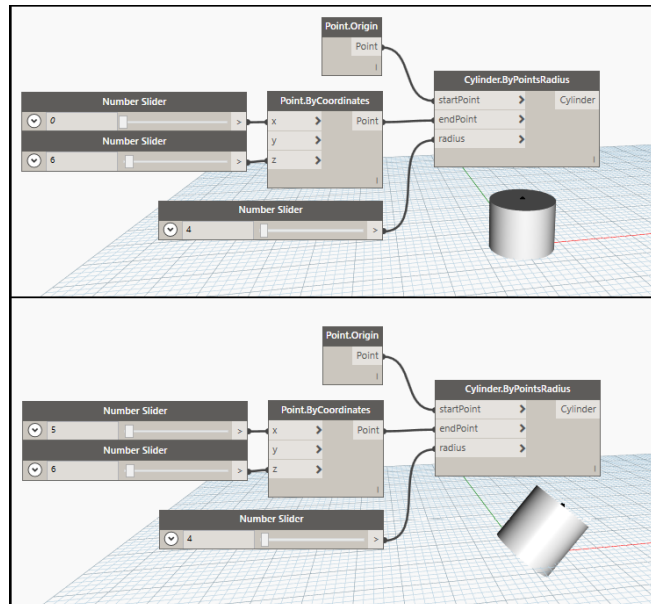


Figura 26 – Manipulação de valores do exercício exemplo para Dynamo

4.4. ESCOLHA DO MÉTODO

Posteriormente ao desenvolvimento dos testes exemplo, descrito nos subcapítulos 4.2 e 4.3, percebeu-se que, apesar dos conhecimentos já adquiridos em programação, em específico na linguagem C#, as dificuldades, para um estudante de engenharia civil desenvolver um *Plug-In* completo, seriam diversas. Os maiores impedimentos prendem-se, essencialmente com dois aspetos: a insuficiência da API do *Revit* e a criação de um *UserForm*. A API do *Revit* é uma interface desenvolvida para programadores experientes, encontra-se pouco modernizada e pouco intuitiva, o que inibe a sua utilização por novos utilizadores. A criação de um *UserForm*, em C#, compatível com a API, de forma a tornar a utilização mais intuitiva, é um trabalho complexo que impossibilitaria a finalização desta dissertação dentro dos prazos previstos. Uma vez que o objetivo do presente documento é sobretudo desenvolver e explorar o método, preferiu-se desenvolver novas capacidades em detrimento do aspeto gráfico, deixando a elaboração do *UserForm* para um desenvolvimento futuro.

O *Dynamo* demonstrou ser uma plataforma intuitiva, de rápida aprendizagem e de comandos simples. Embora não haja tanta informação disponível *online* como a existente sobre o C#, por ser uma linguagem relativamente recente, a *Autodesk* disponibiliza um *site* para troca de informações e dúvidas sobre o *software* (Autodesk 2016). Em abril de 2018, a *Autodesk* lançou a versão 2018.3 do *Revit*, permitindo a criação de *UserForms* para receber *inputs* diretamente do utilizador para o *Dynamo*, através de caixas de diálogo numéricas, alfabéticas ou mesmo botões de decisão (Autodesk 2018a). Versões anteriores ao *Revit* 2018 admitiam *inputs*

oriundos unicamente do *Revit* diretamente para o *Dynamo*. Neste momento é possível aliar as atualizações dos softwares, *Revit*, *Dyno* e *Dynamo* e elaborar *UserForms* automaticamente, de aspeto simples, mas suficientes para uma primeira abordagem.

Apesar das vantagens inerentes, o *software* ainda apresenta algumas incompatibilidades de versões, ao ponto de, em versões mais atuais, existirem funções “*Deprecated*”, ou descontinuadas, em português. Como o próprio nome indica, tratam-se de funções que foram substituídas por outras ou que simplesmente entraram em desuso, complicando o trabalho do programador e diminuindo a interoperabilidade. Assim, a atualização para versões superiores do *Revit*, ou do *Dynamo*, requer uma atualização constante do código e o assíduo tratamento das funções “*Deprecated*” existentes quando o código é transferido entre versões.

Desta forma, a escolha do melhor método recai sobre o *Dynamo*, tornando possível iniciar-se o processo de programação. Afinal, o *Dynamo* foi desenhado pela empresa comercializadora do *Revit* para, especificamente, criar este tipo de pequenos programas.

4.5. PROGRAMAÇÃO DO *PLUG-IN*

A fase de programação em *Dynamo* tem início com a pesquisa de funções que auxiliem ao desenvolvimento das duas fases já apresentadas: a seleção do elemento e a escolha do ensaio a utilizar. Depois da descoberta de algumas funções essenciais como ler o elemento selecionado no *Revit*, selecionar base de dados, manipular listas, escolher entre diferentes opções e apresentar listas, elaborou-se uma pequena base de dados. Para o efeito criou-se uma tabela, em *Excel*, onde foram listados os materiais possíveis e as suas características. A cada material foi associado um elemento selecionável no *Revit* que desencadeia a primeira restrição, na base de dados, por elemento. É de notar que, devido às características do caso de estudo, exposto no capítulo 5 a base de dados apenas foi preenchida com diferentes tipos de betão, contudo estes materiais são exemplificativos, a expansão da base de dados é um processo simples que poderá ser facilmente executado no futuro.

De seguida, deu-se início à leitura da base de dados, pelo *Dynamo*, através da função “*Read From File*” para *Excel*. Este bloco, por si só não lê o ficheiro *Excel*, é necessário adicionar outras funções auxiliares para localizar o ficheiro e a folha de *Excel* que se pretende ler. A Figura 27 exhibe a execução desta parte do código.

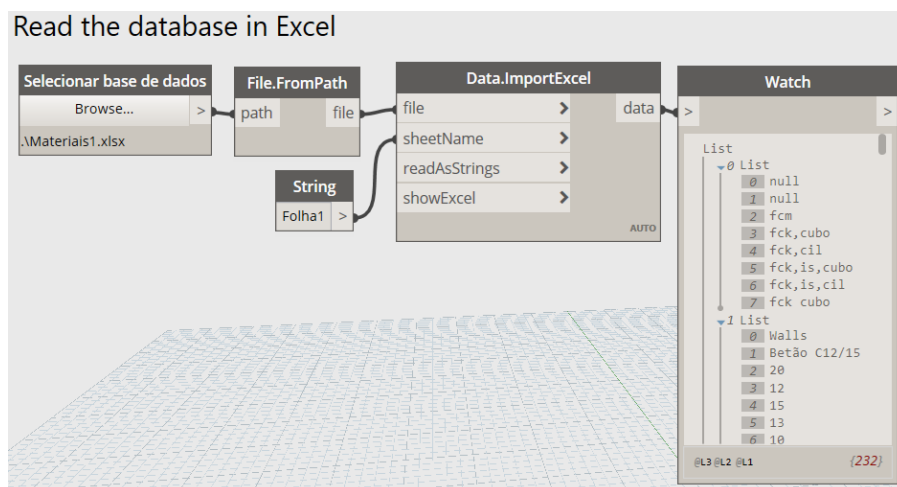


Figura 27 - Leitura da base de dados em Excel

O passo seguinte, exemplificado pela Figura 28, consistiu em seleccionar um elemento do projeto em *Revit* e na leitura das suas características. Com a função “Select Model Element” o *Dynamo* lê, por defeito, o número associado ao elemento, o ID, que por sua vez nos dá acesso a todas as características intrínsecas ao elemento seleccionado. Dessa lista de atributos queremos apenas a “Category” que nos define o elemento por família: Parede, Pilar, Viga ou Laje.

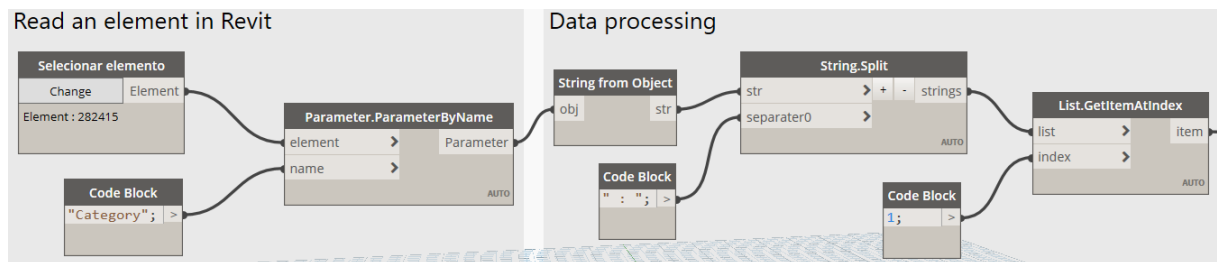


Figura 28 - Leitura do elemento em Revit

Os dados retirados do *Revit* foram tratados e filtrados de modo a serem compatíveis e comparáveis com os elementos da base de dados. De seguida, filtraram-se esses materiais pelo elemento seleccionado obtendo-se a lista provisória. Esta é uma primeira triagem dos materiais possíveis e permite que o utilizador tome o primeiro contacto com os materiais existentes na base de dados.

Para tomadas de decisão, o *Dynamo* apresenta a função “Boolean”: cada pergunta, ou árvore de decisão, é respondida com *True* ou *False*, isto é, a função apenas permite uma resposta binária. Esta forma elementar de tratamento de dados, quando comparada com os ciclos *If*, *For* ou *While* existentes nas linguagens de programação comuns, complica a decisão. É evidente que, neste aspeto, o programa encontra-se pouco desenvolvido e que, os processos de decisão devem ser melhorados nas próximas versões. Para contornar esta situação preferiu-se criar três *Plug-Ins* interligados. Esta opção permite que o utilizador recorra apenas ao *Plug-In* que lhe é mais conveniente, simplificando o processo e evitando as múltiplas escolhas que levariam ao ensaio pretendido.

Assim, o primeiro *Plug-In*, intitulado “Ensaio de Carotes e Provetes”, é constituído por duas perguntas:

- 1º. Pretende introduzir dados de ensaios de carotes ou provetes?
- 2º. O valor introduzido diz respeito a carotes cilíndricos?

Para facilitar a orientação do utilizador inicia-se o programa com a seleção do elemento, no modelo tridimensional, seguido da 1ª pergunta. Se o utilizador responder negativamente será encaminhado para o próximo *Plug-In*, segundo a mensagem: “Abrir o *Plug-In* para leituras do esclerómetro”. Caso o utilizador responda positivamente aparecerá a 2ª pergunta: “O valor introduzido diz respeito a carotes cilíndricos?”. Em caso afirmativo serão recebidos esses valores, em caso negativo serão recebidos valores de carotes ou provetes cúbicos.

Para a obtenção da resistência à compressão característica, quer de carotes cilíndricos, quer de carotes cúbicos, recorreu-se à norma EN 13791. A avaliação da resistência à compressão pode ser realizada por duas alternativas. A alternativa A que permite obter resultados quando

se realiza até 14 provetes e a alternativa B que se aplica quando estão disponíveis 15 ou mais provetes. Contudo, a norma refere ainda que, para a obtenção de uma medição fiável, é necessário ensaiar no mínimo 3 provetes. O método de cálculo da resistência difere entre as duas alternativas, sendo sempre o menor valor possível entre: as equações (1) e (2) da alternativa A, ou as equações (2) e (3) da alternativa B.

$$f_{ck,is} = f_{m(n),is} - k_2 s \quad (1)$$

$$f_{ck,is} = f_{is, \text{ menor}} + 4 \quad (2)$$

$$f_{ck,is} = f_{m(n),is} - k \quad (3)$$

Onde,

$f_{ck,is}$ resistência à compressão *in situ* característica

$f_{m(n),is}$ média dos resultados de n ensaios da resistência à compressão *in situ*

$f_{is, menor}$ menor resultado de ensaio da resistência à compressão *in situ*.

k margem associada ao pequeno número de ensaios, constante no Quadro 2

k_2 coeficiente dependente das disposições no local de utilização ou 1,48

n número de resultados de ensaio

s desvio padrão

Quadro 2 - Margem associada a pequeno número de resultados de ensaio

n	k
10 a 14	5
7 a 9	6
3 a 6	7

O *Dynamo* não permite a introdução de equações, ou seja, o tratamento de dados tem de ser elaborado através das pequenas funções de comparação e requer a utilização de inúmeros blocos. Para simplificar a análise da informação e por contribuir para a clareza do código optou-se por criar um pequeno programa em *Python*, através da função *Python script*. Este código é capaz de, aos dados introduzidos, avaliar que alternativa deve ser utilizada (A ou B), proceder ao cálculo das equações necessárias e apresentar, como *output*, a resistência característica à compressão do betão existente. O código referente a este processo é apresentado no Anexo 1. Na eventualidade de o utilizador introduzir menos de 3 valores surgirá a mensagem de erro: “Deve ensaiar no mínimo 3 provetes para que o valor obtido possa estar de acordo com a norma EN 13791 de 2008”.

De seguida, compara-se o valor obtido com a base de dados e fornecesse ao utilizador a classe do betão. Por este ser um procedimento repetitivo para os dois tipos de carotes, procurou-se otimizar a solução e elaborar um novo *Python Script* que, dependendo do valor recebido filtra a base de dados e apresenta apenas o betão correspondente. A Figura 29 apresenta a sequência que leva à decisão do betão existente e o Anexo 2 apresenta o código em *Python* que filtra os betões da base de dados.

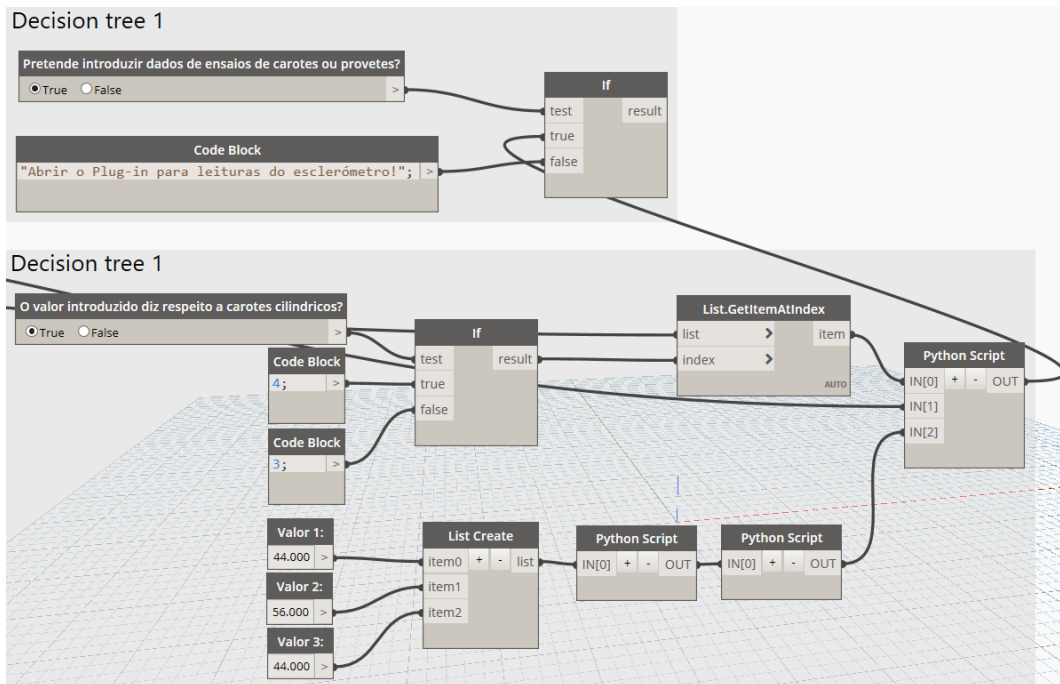


Figura 29 - Blocos de decisão do primeiro Plug-In

O segundo *Plug-In*, intitulado “Ensaio de Esclerómetro”, diz respeito à utilização do esclerómetro de Schmidt e inicia-se com a pergunta:

3º. O esclerómetro está calibrado?

Se o utilizador já tiver o esclerómetro calibrado pode dar continuidade ao *Plug-In*. Caso contrário será encaminhado para o próximo *Plug-In*, segundo a mensagem: "Por favor, calibre o esclerómetro de Schmidt".

O esclerómetro de Schmidt é um instrumento de uso fácil que permite obter um grande número de leituras num curto intervalo de tempo. A avaliação da dureza superficial do betão, através do índice esclerométrico, é relacionado com a resistência à compressão do betão. No entanto, os resultados obtidos são afetados pelos seguintes fatores: Irregularidade superficial; Idade do material testado, Teor de humidade, Tipo de cimento; Carbonatação da superfície do betão; Tipo de agregado; Movimento da peça de betão a ensaiar (Sampaio 2010). Para garantir a maior precisão possível deve-se seguir o procedimento de ensaio da norma NP EN 12504-2 e fazer uma avaliação cuidadosa dos resultados, sendo sempre necessário calibrar o martelo para cada tipo de betão estudado.

Para a avaliação da resistência à compressão característica, dada pelas leituras do esclerómetro, aplica-se a metodologia utilizada na alternativa B da norma EN 13791. Será, então, necessário receber no mínimo 15 valores de resistência que correspondem, segundo a curva de calibração, ao índice de esclerómetro. De acordo com a norma NP EN 12504-2, referente a ensaios não destrutivos em betão, a obtenção do índice esclerométrico traduz-se na mediana de pelo menos 9 leituras válidas, em cada área de ensaio. As leituras estão em conformidade com a norma quando menos de 20% diferir da mediana em mais de 6 unidades. Para facilitar a programação visual, recorreu-se a duas funções *Python script*, uma para o cálculo automático do índice e outra para a identificação do betão de acordo com a base de

dados. Os códigos referentes a estes processos são apresentados nos Anexos 3 e 2, respetivamente.

Assim, para o cálculo do índice esclerométrico e para a obtenção do betão existente recebe-se, pelo menos, 15 conjuntos de 9 leituras: 15 correspondentes às áreas mínimas de ensaio e 9 correspondentes aos valores mínimos de obtenção do índice esclerométrico. Depois dos índices calculados, os valores são comparados com a curva de calibração e obtém-se as resistências correspondentes. Sequencialmente é comparada a resistência com a base de dados, obtendo a classe de betão existente.

Contudo, se o esclerómetro não estiver calibrado será necessário abrir o terceiro *Plug-In*: “Calibração do Esclerómetro”. Este, por sua vez, irá calcular a curva de calibração determinada através da relação entre, pelo menos, 9 pares de resultados (resistências reais das carotes e índices de esclerómetro) conforme a NP EN 13791. À curva base proposta pela norma, representada pelas equações (5) e (6), nos intervalos [20,24] e [24, 50], respetivamente, é imposto o deslocamento necessário para a caracterização da curva de calibração do equipamento. O deslocamento é calculado segundo as equações (7) e (8). Onde, a primeira é a diferença entre a resistência dos provetes e o seu homólogo na curva base e a segunda é a subtração entre a média dos valores de (7) e a multiplicação do fator k_1 pelo desvio padrão da amostra. Mais uma vez foram programadas as funções em *Python* para se poder identificar o valor de Δ_f , responsável pela calibração, e é representado no Anexo 4. O Quadro 3 apresenta os valores possíveis para k_1 , em função do número de pares de resultados, que manifesta ser mais rigoroso quanto menor for o número de áreas em estudo. A curva de calibração é desenhada numa folha de *Excel*. Quando o *Plug-In* termina é apresentada a seguinte mensagem: “Curva de calibração efetuada, por favor verifique o ficheiro Excel”.

$$f_R = 1,25 R - 23 \quad (5)$$

$$f_R = 1,73 R - 34,5 \quad (6)$$

$$\delta_f = f_{is} - f_R \quad (7)$$

$$\Delta_f = \delta f_{m(n)} - k_1 s \quad (8)$$

Onde,

f_{is} resultado do ensaio da resistência à compressão *in situ*

f_R valor inicial da resistência *in situ*, obtida a partir da curva de base do esclerómetro

R resultado do ensaio de esclerómetro

δ_f diferença entre a resistência da carote e o valor da resistência determinada de acordo com a reação base

$\delta f_{m(n)}$ média de n valores de δ_f

Δ_f deslocamento da curva base

k_1 coeficiente dependente do número de pares de ensaio

s desvio padrão da amostra

Quadro 3 - Coeficiente dependente do número de pares de resultados

n	k_1
9	1,67
10	1,62
11	1,58
12	1,55
13	1,52
14	1,50
≥ 15	1,48

A curva de calibração do esclerómetro é, então, paralela à curva base e permite fazer corresponder, de um modo mais fiável, os resultados obtidos pelo aparelho às possíveis resistências à compressão dos ensaios de carotes. Para melhor se compreender que *Plug-In* se deve utilizar, procedeu-se à elaboração do diagrama da Figura 30.

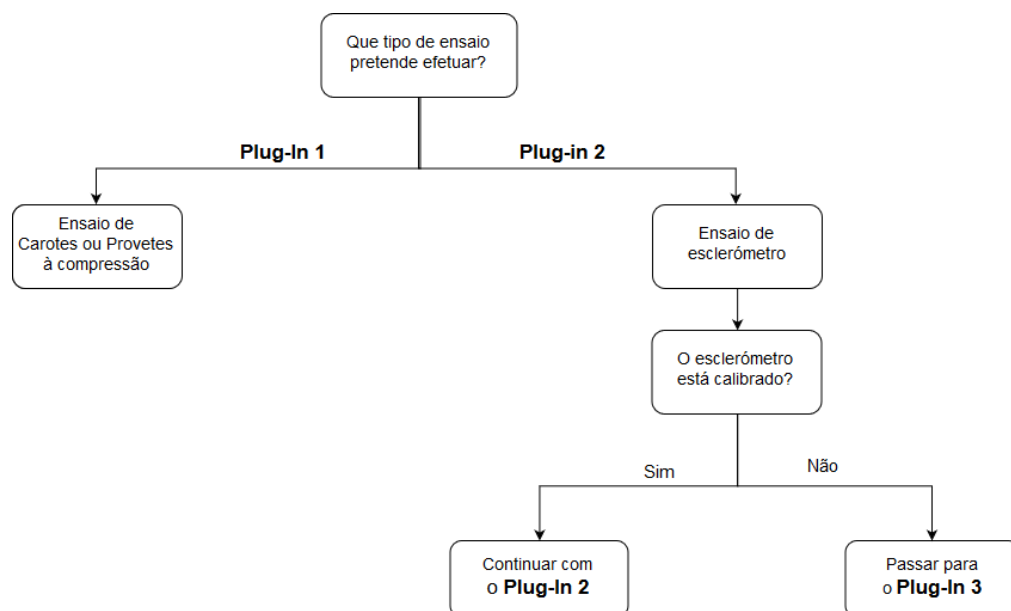


Figura 30 - Diagrama de interligação entre Plug-Ins

A existência de erros foi uma dificuldade sempre presente na criação destes programas, pois, dar início à aprendizagem de uma nova linguagem é sempre um processo lento e moroso. A princípio, foi utilizada a versão 2017 do *Revit*, com o *Dynamo* pré-instalado, na versão 0.9.1, mas depois de se tomar conhecimento das vantagens da versão 2018.3, compatível com o *Dynamo* 2.0.1, optou-se pela atualização do sistema. Esta ação, mediante as inúmeras funções “*Deprecated*” apresentadas, atrasou o desenvolvimento do *Plug-In*. Para ultrapassar alguns inconvenientes do *Dynamo*, recorreu-se ao bloco *Python script*. A função *Python script* utiliza o *IronPython*, uma linguagem de programação *Python*, mas com base em C#. Assim, o programador escreve o código em *Python*, que será traduzido para C#, tornando a função compatível com a API do *Revit*. Contudo o *IronPython* não disponibiliza todas as bibliotecas existentes na linguagem *Python* original.

Embora a nova versão do *Revit* fosse promissora, permitindo a elaboração de um *UserForm* automático, ainda se encontra muito pouco desenvolvida. A elaboração de *UserForms* é efetuada pela ordem de criação dos blocos de *input* no *Dynamo*. Então, se o programador não criou os *inputs* pela ordem desejada, terá de apagar e voltar a criar as funções, pela ordem correta. A utilização do *UserForm* revelou a existência de mais um inconveniente: o *Dynamo* obriga ao preenchimento de todos os campos existentes, não permitindo a escolha do número de ensaios a introduzir. Estes obstáculos podem ser justificados pelo facto de o *Dynamo* ser uma tecnologia recente, e que a implementação de *UserForms* surgiu há apenas 3 meses. Contudo, para se poder prosseguir, optou-se por deixar disponíveis apenas os campos mínimos obrigatórios e compatíveis com o caso de estudo. O código existente no *Dynamo* e no *Python* script já se encontra preparado para receber qualquer número de dados necessários. Assim que o *Dynamo* apresente uma versão mais desenvolvida, é possível atualizar o sistema e utilizar o código no seu potencial máximo.

4.6. DIAGRAMA DE UTILIZAÇÃO DO *PLUG-IN*

Com a evolução da tecnologia e o galopante aumento da informação nela contida, torna-se cada vez mais necessário descrever, com rigor, o modo como um programa funciona, para que seja utilizado da melhor forma possível. Assim, a fase de implementação não atenta apenas no ato de executar o *Plug-In*. É importante apresentar, de uma forma simplificada, específica e intuitiva, o trabalho realizado. A UML – *Unified Modelling Language* responde ao desafio proposto, garantindo que, de uma forma visual, a informação é corretamente transmitida. É utilizada no quotidiano de empresas de natureza tecnológica, que irão construir um sistema informático, ajudando a dominar o fluxo de informação entre os gestores, a equipa de desenvolvimento e os utilizadores (Nunes e O'Neill 2004). Esta linguagem é também usada para documentar, ao longo de todo o ciclo de vida, o desenvolvimento do programa, dando início à tarefa de análise até à manutenção do sistema.

Para podermos escolher o diagrama que melhor se ajusta ao trabalho desenvolvido, a UML disponibiliza o seguinte conjunto (Nunes e O'Neill 2004):

- Diagrama de *Use Case*: Identifica as fronteiras do sistema e descreve os serviços que devem ser disponibilizado a cada utilizador. É ideal para programas em que cada utilizador tem acesso a serviços diferentes.
- Diagrama de Classes: Descreve a estrutura de informação. É muito utilizado para sistemas programados em linguagens orientadas a objetos.
- Diagrama de Objetos: Descreve a identidade, os relacionamentos, os atributos e as operações de cada objeto. É vulgarmente utilizado como um complemento ao diagrama de classes, mas pode ser utilizado individualmente.
- Diagrama de Interação: Ilustra a interação entre os objetos e o sistema.
- Diagrama de Atividade: Realça o encadeamento de cada objeto, numa ótica de simplificar a sequência de passos necessário à automatização do processo. É utilizado para descrever cada um dos *use case*.
- Diagrama de Estados: Descreve as alterações do comportamento dos objetos em resultado da ocorrência de certos eventos.

- Diagrama de Componentes: Realça a arquitetura de uma aplicação informática em termos de componentes de software. É ideal para identificar que componente está a ser utilizada ou subcarregada.
- Diagrama de Instalação: Ilustra a fisiologia do equipamento informático de acordo com a interação entre os diversos equipamentos. É utilizado como complemento ao diagrama de componentes.

Para esta dissertação optou-se pelo diagrama de interação visto ser uma união entre o diagrama de sequência, que realça a ordem cronológica das mensagens enviadas e recebidas entre objetos, e o diagrama de colaboração, que destaca a organização estrutural dos objetos que trocam mensagens. A notação normalizada é retratada na Figura 31.

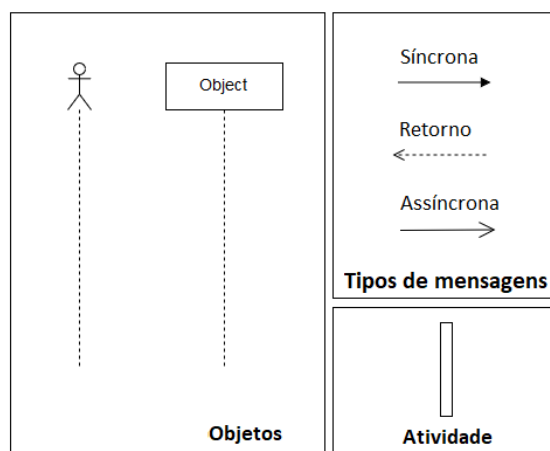


Figura 31 - Notação utilizada para desenvolver o diagrama de interação

O ator, facilmente identificado pelo *stickman*, é um objeto que requer uma aparência diferenciada por simbolizar o utilizador que, regularmente, inaugura o diagrama, dando início à ação desenvolvida pelo *Plug-In*. Os objetos identificados neste trabalho: Utilizador, *Revit*, *Dynamo* e Base de Dados, trocam mensagens entre si por ação do utilizador. A cada objeto é associada uma linha temporal que acompanhará o seu ciclo de atividade, nesse intervalo o objeto assume o controlo, enviando e/ou recebendo dados de outro objeto. As mensagens trocadas espelham a invocação de um serviço, a passagem de informação ou a receção de dados, e podem ser do tipo síncrono, assíncrono, simples ou de retorno. Mensagens síncronas significam que o emissor necessita de uma resposta do recetor. Uma mensagem assíncrona revela que o objeto emissor prossegue o seu procedimento e mantém o controlo sem aguardar feedback. Uma mensagem de retorno retrata a resposta dada pelo recetor ao emissor, portanto é uma sequência da mensagem síncrona. A Figura 32 ilustra o diagrama de interação elaborado para utilização do *Plug-In*.

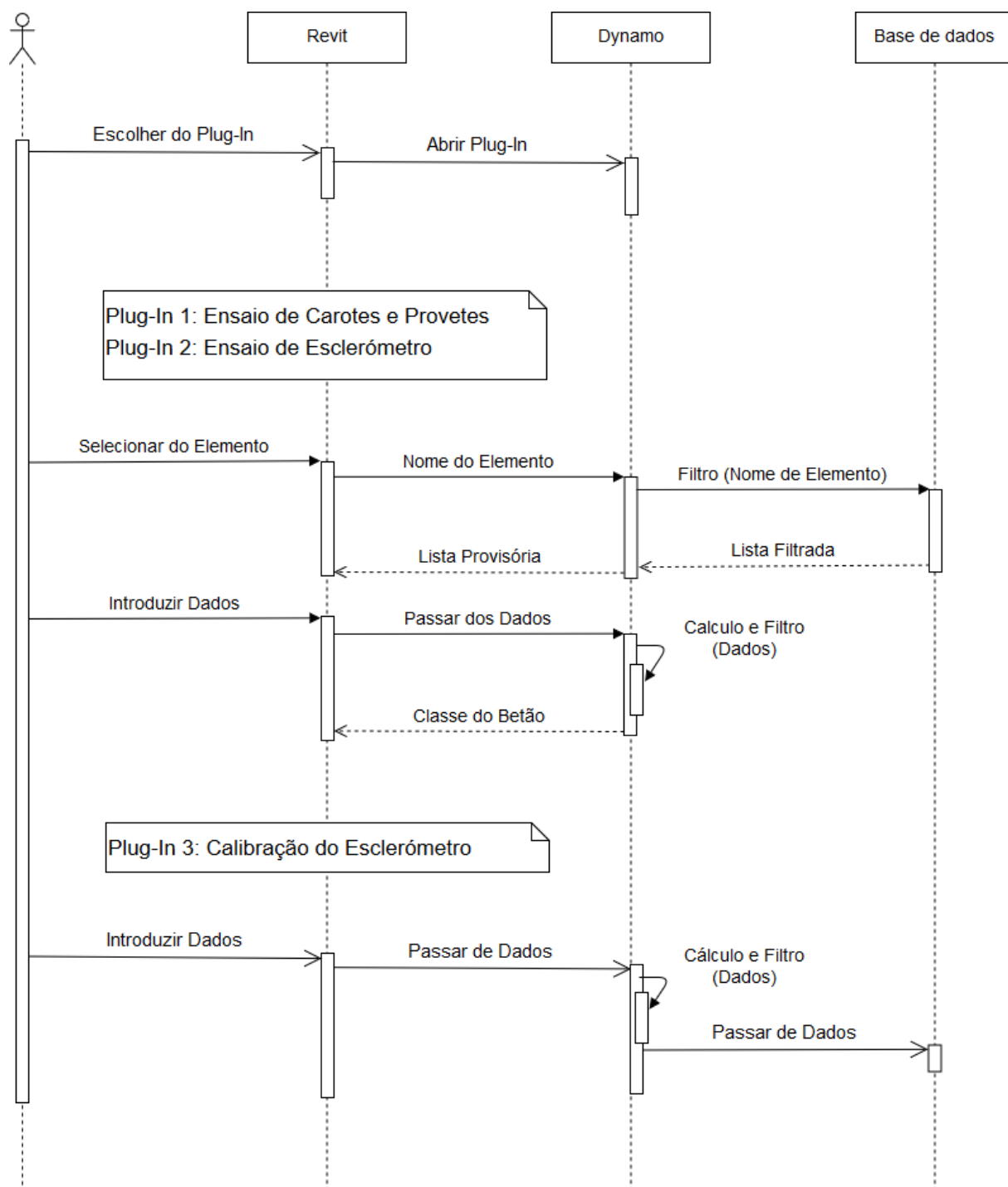


Figura 32 – Digrama de interação UML

5

CASO DE ESTUDO

5.1. DESCRIÇÃO

O caso de estudo tem por base um relatório técnico de “avaliação de ensaio de caracterização dos materiais e avaliação da segurança das lajes de piso de uma construção existente”, realizada em maio de 2014.

A estrutura, ainda em construção, é uma moradia constituída por três pisos: cave, rés-do-chão e primeiro piso. A cave é constituída por 14 pilares, 3 muros de betão armado e 3 painéis de laje com 24 cm de espessura. O rés-do-chão, por sua vez, possui 14 pilares, onde apenas 10 são a sequência dos pilares inferiores e 4 são novos pilares descentrados, apoiados na laje inferior, ou com diferente geometria, e 3 painéis de laje com 23 cm de espessura. O primeiro piso apresenta os 14 pilares na totalidade, mas também aqui existem 3 pilares dissociados e 4 painéis de laje com 18 cm de espessura. Os muros e os pilares são em Betão armado, as lajes são do tipo pré-fabricado com vigotas do fabricante *Pavinorte*. Verificou-se que a estrutura não estava em conformidade com os desenhos do projeto pois estes referiam que, apenas a laje inferior do piso 0 era de vigotas, as restantes, piso 1 e cobertura, seriam maciças com 30 cm e 20 cm de espessura, respetivamente. Estas apoiar-se-iam em paredes de betão armado, que não existem na construção atual. Os materiais previstos foram o betão da classe de resistência C20/25 e o aço de classe A400. Como é evidente, a estrutura não foi executada de acordo com o projeto de estabilidade, sendo totalmente distinta da que consta dos documentos de licenciamento. Posto isto, a estrutura atual foi modelada em *Revit*, como se apresenta na Figura 33, com vista a servir de exemplo para a aplicação do *Plug-In*.

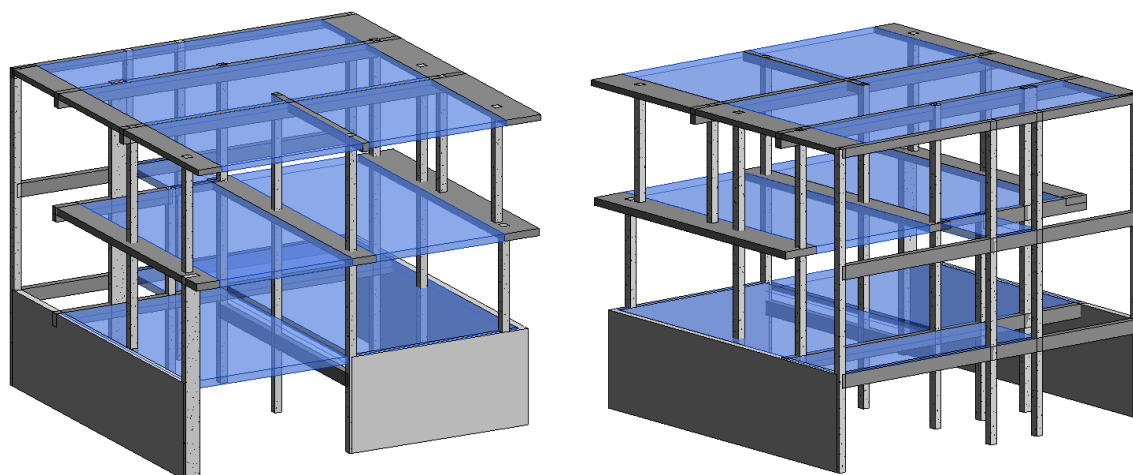


Figura 33 - Modelo tridimensional da estrutura em Revit

Constatou-se também uma série de deficiências graves relacionadas com a conceção, execução e estado geral da estrutura. O relatório descreve a análise da estrutura através do levantamento estrutural, da caracterização do betão existente e da avaliação de segurança. O foco desta dissertação recai na caracterização do betão uma vez que o objetivo é identificar os materiais existentes em obra. Contudo, sumariamente, concluiu-se, através dos parâmetros apresentados no Quadro 4, que as lajes do piso 0 e 1 e a generalidade das vigas não verificam as condições mínimas de segurança.

Quadro 4 - Síntese dos parâmetros de avaliação

Parâmetros	Lajes	Vigas
Resistência à flexão	Não satisfeita Cerca de 35% inferior à necessária	Não satisfeita
Resistência ao corte	Não satisfeita	Não satisfeita
Disposições construtivas	Satisfeita Transferência de força das vigotas às vigas Amarração de cabos de pré-esforço Disposição da armadura adicionais Disposição de armaduras adicionais entre blocos de aligeiramento	Não satisfeita
Qualidade do betão	Muito fraca Aderência entre os fios de pré-esforço é questionável	Muito fraca
Qualidade da construção	Baixa	Baixa

Ainda assim, devido à baixa qualidade da construção, não existem garantias que todas as regras especificadas na NP EN 1992-1-1 sejam cumpridas. Acrescem problemas relevantes de durabilidade e segurança. A durabilidade é relativa à corrosão da armadura, devido à elevada porosidade do betão e à quase inexistência de recobrimento, torna-se difícil assegurar o funcionamento solidário dos dois materiais aço e betão. As preocupações de segurança devem-se ao facto de esta estrutura não conter elementos de contraventamento, uma vez que os pilares se revelam insuficientes para resistir aos momentos de continuidade com as vigas, mesmo os decorrentes de ações gravíticas.

Numa das visitas à obra foram retirados 7 carotes de betão, 4 em vigas e 3 em pilares. Devido à fraca resistência do material, duas das amostras, uma de viga e outra de pilar, partiram durante a retificação. Os carotes intactos foram ensaiados à compressão uniaxial, foi medida a velocidade de propagação de ultrassons e foi avaliada a massa volúmica no laboratório de estruturas da FEUP. Os resultados destes ensaios podem ser consultados na Quadro 5 e nos boletins de ensaio do Anexo 5.

Quadro 5 - Resultado de ensaios a carotes

Referência	Resistência à Compressão (MPa)	Média	Massa Volúmica (Kg/m ³)	Média	Velocidade (Km/s)
Pilar – P1F	8.8	9.7	2238	2197	3.39
Pilar – P6F	9.70		2182		3.29
Pilar – P5E	10.7		2171		3.32
Viga – V1B	5.46	7.6	2205	2174	3.1
Viga – V1F	9.55		2143		3.39

Pôde-se concluir que a média da massa volúmica, quer nos pilares, quer nas vigas é significativamente inferior aos 2400 – 2500 Kg/m³ utilizados nos betões convencionais. Este índice traduz-se numa elevada porosidade do betão levando a perdas de durabilidade e a baixas resistências. Antes do ensaio à compressão, foi ainda medida a velocidade de propagação dos ultrassons onde se obteve, em todos os provetes, velocidades inferiores ao valor de referência mínimo de 4 Km/s previsto pela NP EN 13791. Seguiu-se, então, o ensaio da resistência à compressão que, devido ao número reduzido de carotes não foi possível calcular a resistência seguindo os processos normalizados. Mesmo assim os resultados médios obtidos pelos carotes existentes apontam para um betão de resistência inferior à da classe C12/15 e muito díspar da classe C20/25, estipulada pelo projeto inicial.

Para complementar os dados dos carotes realizaram-se ensaios para obtenção da resistência superficial com recurso ao esclerómetro de Schmidt. Aqui foram efetuados ensaios a 16 Pilares e a 2 vigas. Seguindo a norma NP EN 13791, foram efetuados no mínimo 9 leituras em cada elemento.

5.2. UTILIZAÇÃO DO *PLUG-IN*

Depois das tarefas de planeamento, análise e programação chega a altura de se proceder à implementação do plug-in. Quando iniciado, o *Revit* carrega os *Plug-Ins* e mantém-nos à disposição do utilizador, para que quando for necessário, estes possam ser executados. No menu *Manage*, ou Gerenciar em Português, é apresentado o separador de programação visual que contem as ligações para o *Dynamo* e para o *Dynamo Player*. Para abrir o *Plug-In* e inspecionar o seu código deve-se abrir o *Dynamo*. Como se pretende por em prática o *Plug-In* com os dados do caso de estudo abre-se a caixa de diálogo do *Dynamo Player*, apresentada na Figura 34.

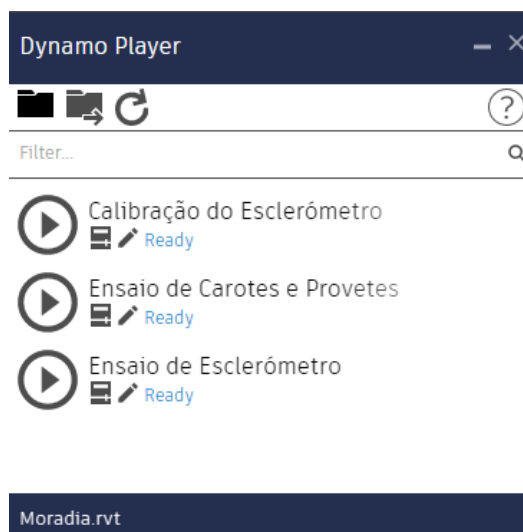


Figura 34 – Aparência do Dynamo Player

Iniciando com a calibração do esclerômetro, abre-se o *UserForm* e, por apenas estar disponível o número mínimo de valores de ensaio, o Dynamo associa automaticamente esses valores aos respectivos espaços. É pedido para confirmar os valores dos 10 grupos apresentados, os primeiros 9 grupos, compostos por 9 espaços cada, dizem respeito às 9 leituras do esclerômetro. No último grupo, também composto por 9 valores, deve-se introduzir a resistência obtida nos 9 ensaios de carotes. Uma vez que dispomos apenas de dois pilares com os dois ensaios, os restantes valores da resistência à compressão foram arbitrados e associados a valores reais do esclerômetro. Os valores de teste do plug-in são apresentados no Quadro 6 e a curva de calibração é ilustrada na Figura 35.

Quadro 6 – Resistência dos carotes e leituras correspondentes

Referência	Posição	Resistência das Carotes (MPa)	Leituras realizadas								
Pilar – P3F	Piso 1	8,8	42	38	45	42	38	42	34	38	45
Pilar – P6F	Piso 1	9,7	26	26	26	27	27	27	27	27	27
Pilar – P5E	Piso 1	10,7	28	28	30	30	34	30	36	31	34
Pilar – P5D	Piso 1	11,2	24	26	36	32	29	26	36	26	25
Pilar – P5C	Piso 1	10,2	26	28	28	28	27	26	28	27	26
Pilar – P1C	Piso 1	8,8	28	28	28	30	31	29	32	36	28
Pilar – P1-2 AB	Cobertura	9,3	34	28	40	28	30	34	30	32	30
Pilar – P4 AB	R/C	11,1	32	30	30	28	30	28	30	30	28
Pilar – P5C	R/C	9,3	29	31	30	31	26	32	28	34	32

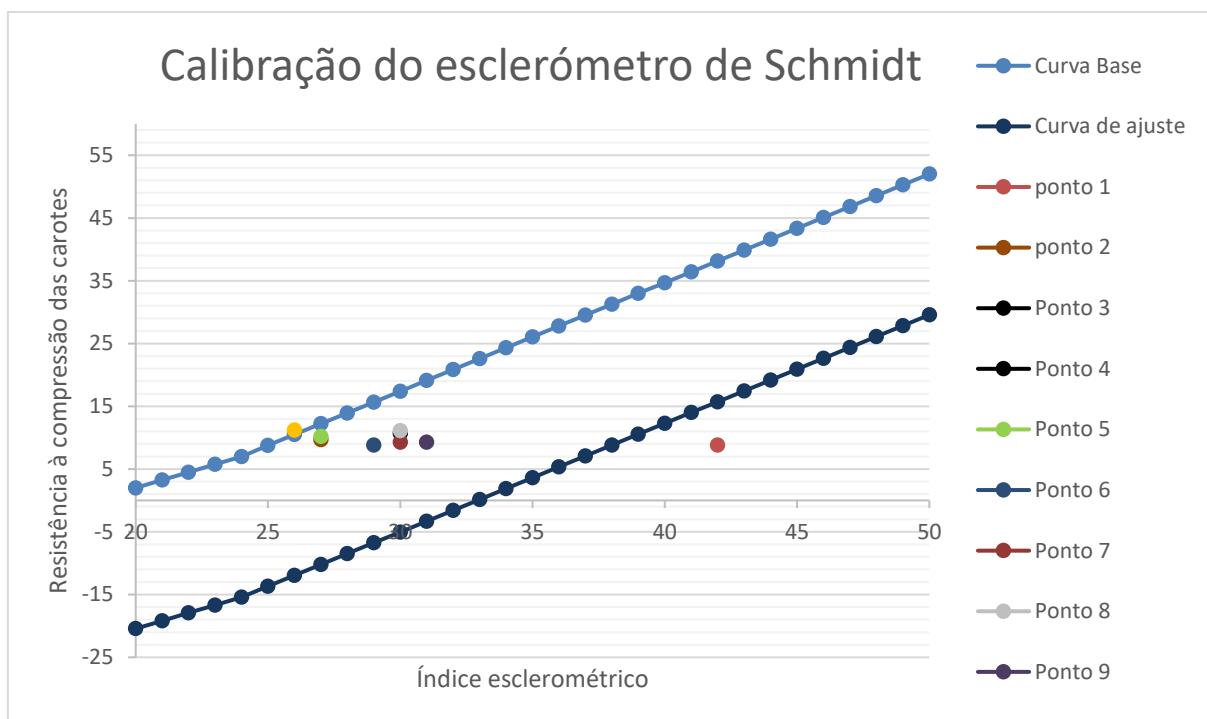


Figura 35 - Curva de calibração do esclerómetro

Depois do esclerómetro calibrado, é possível utilizar as leituras do esclerómetro para obter a classe do betão existente em obra. Escolhe-se, então, o Plug-In “Ensaio de Esclerómetro” e introduz-se os 15 conjuntos de valores do Quadro 7.

Quadro 7 - Leituras do esclerómetro

Referência	Posição	Leituras realizadas								
Pilar – P3F	Piso 1	42	38	45	42	38	42	34	38	45
Pilar – P6F	Piso 1	26	26	26	27	27	27	27	27	27
Pilar – P5E	Piso 1	28	28	30	30	34	30	36	31	34
Pilar – P5D	Piso 1	24	26	36	32	29	26	36	26	25
Pilar – P5C	Piso 1	26	28	28	28	27	26	28	27	26
Pilar – P1C	Piso 1	28	28	28	30	31	29	32	36	28
Pilar – P1-2 AB	Cobertura	34	28	40	28	30	34	30	32	30
Pilar – P4 AB	R/C	32	30	30	28	30	28	30	30	28
Pilar – P5C	R/C	29	31	30	31	26	32	28	34	32
Pilar – P3-4 C	R/C	32	32	28	32	36	36	32	30	34
Pilar – P5C	Cobertura	32	36	34	32	36	36	32	32	32
Pilar – P4C	Cobertura	32	26	24	30	30	28	30	26	26
Pilar – P3B	Cobertura	26	26	28	26	26	26	26	26	26
Pilar – P4C	Piso 1	26	30	26	25	24	30	26	24	24
Pilar – 1F	Piso 1	28	30	26	26	30	24	24	30	24

Surge, por fim, a classe de betão correspondente: Inferior a C12/15.

Segue-se a utilização do *Plug-In* para ensaio de carotes e provetes. O *UserForm* correspondente é apresentado no Anexo 6 e os valores de ensaio são apresentados na Quadro 8.

Quadro 8 - Resistência à compressão de carotes

Referência	Resistência à Compressão (MPa)
Pilar – P1F	8.8
Pilar – P6F	9.70
Pilar – P5E	10.7

Obtém-se, então, a classe de resistência correspondente: Inferior a C12/15.

Face ao exposto, e em concordância com o relatório técnico, conclui-se que o betão existente apresenta muito fraca qualidade.

A baixa qualidade foi detetada na visita à obra e comprovada pelos ensaios efetuados. O relatório técnico destaca a fragilidade dos carotes e o destacamento da camada de recobrimento das armaduras, deixando visível os estribos e algumas armaduras longitudinais. Os ensaios efetuados durante a avaliação permitiram constatar não só a reduzida resistência do betão, mas também, a elevada porosidade do material. A existência conjunta destes fatores acelera os problemas de corrosão e, com a passagem do tempo, irá reduzir a resistência e a segurança global da estrutura.

A automatização do processo de tratamento dos dados de ensaios desenvolvida no decurso desta dissertação permitiu acelerar a resposta e minimizar o erro humano. Através das classes de betão assim identificadas conclui-se que o betão presente em obra não está em conformidade com o projeto nem com a regulamentação em vigor.

6

CONCLUSÕES E DESENVOLVIMENTOS FUTUROS

6.1. CONCLUSÃO

Numa primeira análise deste trabalho pode-se concluir que todos os objetivos foram cumpridos tendo-se realizado as quatro etapas propostas inicialmente.

Foi elaborado um modelo tridimensional e paramétrico, utilizando o programa *Revit* e recorrendo à geometria bem definida do caso de estudo. Nesse modelo foi possível adicionar novas famílias para representação, quer dos provetes, quer das áreas de leitura do esclerómetro.

Desenvolveu-se um *Plug-In* que permite classificar o betão e calibrar o esclerómetro de Schmidt. A filtragem dos betões existentes, na base de dados, contou com a programação em *Python* que, ao elemento selecionado no modelo, retira os dados aí armazenados para calcular a resistência. A programação visual, em *Dynamo*, e a criação de código em *Python*, foi, sem dúvida, a tarefa mais desafiante. O *Dynamo* apresenta, ainda, algumas limitações, mesmo em tarefas simples como introdução de funções ou processos de decisão. Mas, recorrendo à programação em *Python*, foi possível contornar os inconvenientes encontrados. É importante lembrar que, o *Dynamo*, se trata de uma tecnologia recente, com apenas dois anos, e algumas das funcionalidades utilizadas, como o *UserForm*, foram lançadas para o mercado há cerca de dois meses.

A avaliação da resistência característica à compressão do betão, foi calculada por ensaios de provetes/carotes e por leituras do esclerómetro, utilizando as normas NP EN 13791 e EN1504-2. Contudo, cabe ao engenheiro avaliar as exceções que constam da NP EN 13791, capítulo 9, intitulado “avaliação em caso de dúvida sobre a conformidade do betão baseada em ensaios normalizados”.

É então possível incluir no modelo BIM informação sobre a caracterização da resistência do betão existente, através dos seguintes processos:

- 1º. Elaborar um modelo tridimensional, segundo a metodologia BIM;
- 2º. Escolher da área de ensaio e identificar a sua posição no modelo;
- 3º. Armazenar a informação dos ensaios no modelo;

- 4º. Utilizar o *Plug-In* para obter instantaneamente a classe do betão existente;
- 5º. Armazenar esta informação para memória futura.

Esta experiência permitiu à autora adquirir novos conhecimentos e desenvolver outros que contribuirão para o seu futuro como profissional de engenharia civil.

6.2. DESENVOLVIMENTOS FUTUROS

Como desenvolvimentos futuros, e com o objetivo de aumentar e melhorar as funcionalidades do *Plug-In*, sugerem-se os seguintes trabalhos:

A introdução de outras técnicas de ensaio, não-destrutivas como a força de arranque (*Pull out test*), velocidade de propagação dos ultrassons e arranque por tração direta, (*Pull-off test*), regulados pelas normas NP EN 12504-3, NP EN 12504-4, NP EN 1542 e NP EN 13791.

A evolução da interface de ligação utilizador/*Plug-In*, através do desenvolvimento de um *UserForm* mais prático e atrativo. Uma vez que o código já se encontra desenvolvido para o tratamento de qualquer número de ensaios, é, apenas, necessário capacitar o *UserForm* com uma *input box* que receba o número de ensaios que se pretende introduzir.

BIBLIOGRAFIA

- AIA. 2013. G202-2013 Project BIM Protocol Form.
- Arayici, Coates, Koskela, Kagioglou, Usher e O'Reilly, K. 2011. "Technology adoption in the BIM implementation for lean architectural practice." *Automation in Construction* 20 (2):189-95. doi: 10.1016/j.autcon.2010.09.016.
- Autodesk. 2016. "Dynamo." accessed 24-04-2018. <http://dynamobim.org/>.
- Autodesk. 2017. "Dynamo Dictionary." <http://dictionary.Dynamobim.com/#/>.
- Autodesk. 2018a. "Autodesk Revit 2018 Product Updates." <https://knowledge.autodesk.com/support/revit-products/downloads/caas/downloads/content/autodesk-revit-2018-product-updates.html>.
- Autodesk. 2018b. The Dynamo Primer.
- Autodesk. 2018c. "My First Revit Plug-in Overview." accessed 22-04-2018. <https://knowledge.autodesk.com/support/revit-products/learn-explore/caas/simplecontent/content/my-first-revit-plug-overview.html>.
- Azhar, S. 2011. "Building information modeling (BIM): Trends, benefits, risks, and challenges for the AEC industry." *Leadership and management in engineering* 11 (3):241-252.
- Bakowski, J. 2017. "Analytical tools for functional assessment of architectural layouts." World Multidisciplinary Civil Engineering-Architecture-Urban Planning Symposium (WMCAUS), 12-16 June 2017, UK.
- Bergin, M. 2012. "A Brief History of BIM ", accessed 04-03-2018. <https://www.archdaily.com/302490/a-brief-history-of-bim>.
- Biljecki, Ledoux e Stoter, J. 2016. "An improved LOD specification for 3D building models." *Computers, Environment and Urban Systems* 59:25-37. doi: <https://doi.org/10.1016/j.compenvurbsys.2016.04.005>.
- BIMForum. 2017. Level of Development Specification Part I.
- Bouška, R. 2016. "Evaluation of Maturity of BIM Tools across Different Software Platforms." *Procedia Engineering* 164:481-486. doi: <https://doi.org/10.1016/j.proeng.2016.11.648>.
- Bruno, De Fino e Fatiguso, F. 2018. "Historic Building Information Modelling: performance assessment for diagnosis-aided information modelling and management." *Automation in Construction* 86:256-276. doi: <https://doi.org/10.1016/j.autcon.2017.11.009>.
- BuildingSMART. 2013. "Related Specifications Summary ", accessed 06-03-2018. <http://www.buildingsmart-tech.org/specifications/related-specifications>.
- Cabinet Office, . 2011. Government Construction Strategy
- CAD, N.J.H. 2017. "BIM." accessed 20-03-2018. http://njhcadservices.co.uk/?page_id=32.
- Comissão Europeia, UE. 2014. Energia.
- Construction, M. 2014. The Business Value of BIM For Construction in Major Global Markets: how contractors around the world are driving innovation with building information modeling. In *Smart MarketReport*.

- Dawood, Sriprasert, Mallasi e Hobbs, B. 2003. "Development of an integrated information resource base for 4D/VR construction processes simulation." *Automation in Construction* 12 (3):123-31. doi: 10.1016/S0926-5805(02)00045-6.
- Eastman, C. 1975. "The use of computers instead of drawings in building design." *AIA Journal* 63 (3):46-50.
- Eastman, Teicholz e Sacks, R. 2011. *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*: John Wiley & Sons.
- Engelbart, D. 2001. "Augmenting human intellect: a conceptual framework." PACKER, Randall and JORDAN, Ken. *Multimedia. From Wagner to Virtual Reality*. New York: WW Norton & Company:64-90.
- Ferreira, B. 2015. "Desenvolvimento de metodologias BIM de apoio aos trabalhos construtivos de medição e orçamentação." Dissertação de Mestrado, FEUP.
- Hill, McGraw. 2007. "Interoperability in the construction industry." *Smartmarket report*.
- Hosseini, Martek, Zavadskas, Aibinu, Arashpour e Chileshe, N. 2018. "Critical evaluation of off-site construction research: A Scientometric analysis." *Automation in Construction* 87:235-247. doi: <https://doi.org/10.1016/j.autcon.2017.12.002>.
- INE. 2017. "Educação e formação de adultos em Portugal: retrato estatístico de uma década." accessed 20-04-2018. https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaques&DESTAQUESdestbou=281447960&DESTAQUESmodo=2.
- Jernigan, F. 2007. "BIG BIM little bim." *The practical approach to Building*.
- Jiang, X. 2017. "Research on Application of BIM 5D Technology in Central Grand Project." *Procedia Engineering* 174:600-10. doi: 10.1016/j.proeng.2017.01.194.
- Jongeling e Olofsson, T. 2007. "A method for planning of work-flow by combined use of location-based scheduling and 4D CAD." *Automation in Construction* 16 (2):189-98. doi: 10.1016/j.autcon.2006.04.001.
- Kamardeen, I. 2010. "8D BIM modelling tool for accident prevention through design." 26th Annual ARCOM Conference.
- Krona, M. 2016. "Why 4D, 5D and 6D BIM need real manufacturers' BIM objects hosted on a professional cloud solution." accessed 2-04-2018. <https://bimobject.com/en/blog/post/why-4d-5d-and-6d-bim-need-real-manufacturers-bim-objects-hosted-on-a-professional-cloud-solution>.
- Manziona, L. 2016. "BIM Nível 2: realidade no Reino Unido." accessed 1/01/2018. <http://www.coordenar.com.br/bim-nivel-2-realidade-no-reino-unido/>.
- Mayo, J. 2016a. "Lesson 8: Class Inheritance." accessed 20-05-2018. <https://csharp-station.com/Tutorial/CSharp/Lesson08>.
- mayo, J. 2016b. "Lesson 9: Polymorphism." accessed 20-05-2018. <https://csharp-station.com/Tutorial/CSharp/Lesson09>.
- Mayo, J. 2016c. "Lesson 19: Encapsulation." accessed 20-05-2018. <https://csharp-station.com/Tutorial/CSharp/Lesson19>.
- Meira, A. 2016. "Desenvolvimento de ferramentas compatíveis com modulação em BIM para análise do custo da habitação social." Dissertação de Mestrado, FEUP.
- Menezes, G. 2012. Breve histórico de implantação da plataforma BIM. In *Cadernos de Arquitetura e Urbanismo*.
- Microsoft. 2017. "The history of C#." <https://docs.microsoft.com/en-us/dotnet/csharp/whats-new/csharp-version-history>.
- Microsoft. 2018. "Get started with the .NET Framework." <https://docs.microsoft.com/en-us/dotnet/framework/get-started/>.
- Monteiro e Poças Martins, J. 2013. "A survey on modeling guidelines for quantity takeoff-oriented BIM-based design." *Automation in Construction* 35:238-253.
- Nederveen e Tolman, F. 1992. "Modelling multiple views on buildings." *Automation in Construction* 1 (3):215-224.

- NIBS. 2007. United States National Building Information Modeling Standard - Version 1: Overview, Principles, and Methodologies.
- Nical e Wodynski, W. 2016. "Enhancing Facility Management through BIM 6D." *Procedia Engineering* 164:299-306. doi: 10.1016/j.proeng.2016.11.623.
- Nunes e O'Neill, H. 2004. *Fundamental de UML*.
- Pelipenko e Gogina, E. 2017. "Real application of BIM in the engineering system design for energy management." *Energy Management of Municipal Transportation Facilities and Transport (EMMFT 2017)*, 10-13 April 2017, UK.
- Python. 2018. "Beginner's Guide to Python."
- Qiqi, Jongsung e Cheng 2016. "A financial decision making framework for construction projects based on 5D Building Information Modeling (BIM)." *International Journal of Project Management* 34 (1):3-21. doi: 10.1016/j.ijproman.2015.09.004.
- Reece, J. 2002. "Behind the scenes at rize: A profile of co-founder Leonid Raiz." accessed 02-03-2018. <http://www.rize3d.com/blog/behind-scenes-rize-profile-co-founder-leonid-raiz>.
- Richard. 2010. "Browser Plugins vs Extensions – the difference." accessed 24-04-2018. <http://colonelpanic.net/2010/08/browser-plugins-vs-extensions-the-difference/>.
- Sampaio, C. 2010. "Caracterização do betão através de técnicas não-destrutivas."
- Sanhudo e Poças Martins, J. 2016. "Desenvolvimento de um Software para a Determinação e Mitigação do Escoamento Superficial."
- Santos, J. 2006. "Algoritmia e Programação." Sebenta, Departamento de Engenharia Informática, ISEP.
- Silva, J. 2013. "Princípios para o desenvolvimento de projetos com recurso a ferramentas BIM." Dissertação de Mestrado, FEUP.
- Smith e Tardif, M. 2009. *Building information modeling: a strategic implementation guide for architects, engineers, constructors, and real estate asset managers*: John Wiley & Sons.
- Standard, National BIM. 2015. "buildingSMART alliance Releases NBIMS-US™ Version 3." accessed 26-03-2018. <https://www.nationalbimstandard.org/buildingSMART-alliance-Releases-NBIMS-US-Version-3>.
- Sullivan, Barlish e. 2012. "How to measure the benefits of BIM — A case study approach." *Automation in Construction* 24:149-159. doi: <https://doi.org/10.1016/j.autcon.2012.02.008>.
- Venâncio, M. 2015. "Avaliação da Implementação de BIM-Building Information Modeling em Portugal." Dissertação de Mestrado, FEUP.

ANEXOS

ANEXO 1

Objetivo: Avaliação da resistência à compressão *in situ* característica por ensaios em carotes.

Linguagem de programação: *Python*.

Enable Python support and load DesignScript library

```
import clr
clr.AddReference("RevitAPI")
from Autodesk.Revit.DB import *
clr.AddReference("RevitServices")
import RevitServices
from RevitServices.Persistence import DocumentManager
from RevitServices.Transactions import TransactionManager
doc = DocumentManager.Instance.CurrentDBDocument
```

#Definir funções de media e desvio padrão

```
def mean(data):
    """Return the sample arithmetic mean of data."""
    n = len(data)
    if n < 1:
        raise ValueError('mean requires at least one data point')
    return sum(data)/float(n) # in Python 2 use sum(data)/float(n)
```

```
def _ss(data):
    """Return sum of square deviations of sequence data."""
    c = mean(data)
    ss = sum((x-c)**2 for x in data)
    return ss
```

```
def stddev(data, ddof=1):
    """Calculates the population standard deviation
    by default; specify ddof=1 to compute the sample
    standard deviation."""
    n = len(data)
    if n < 2:
        raise ValueError('variance requires at least two data points')
    ss = _ss(data)
    pvar = ss/(n-ddof)
    return pvar**0.5
```

Place your code below this line

#Definir Variáveis

```
listavg = IN[0]
n = len(listavg)
avg=0
desviopa=0
minimo=min(listavg)
k=0

avg=mean(listavg)
desviopa= stddev(listavg, 1)
```

#Definir Abordagem B

```
if n>=3 and n<=14:
```

```
    #Definir K
```

```
    if n>=3 and n<=6:
```

```
        k=7
```

```
    if n>6 and n<=9:
```

```
        k=6
```

```
    if n>9 and n<=14:
```

```
        k=5
```

```
    #Definir output
```

```
    if ((avg - k) < (minimo +4)):
```

```
        OUT=avg-k
```

```
    if ((avg - k) >= (minimo +4)):
```

```
        OUT=minimo+4
```

#Definir Abordagem A, considerando $K_2 = 1,48$

```
if n>=15:
```

```
    #Definir output
```

```
    if ((avg - 1.48 * desviopa) < (minimo +4)):
```

```
        OUT=avg - 1.48 * desviopa
```

```
    if ((avg - 1.48 * desviopa) >= (minimo +4)):
```

```
        OUT=minimo+4
```


ANEXO 2

Objetivo: Comparação do valor da resistência com a base de dados, obtendo a classe de betão existente.

Linguagem de programação: *Python*.

```
# Enable Python support and load DesignScript library
import clr
clr.AddReference('ProtoGeometry')
from Autodesk.DesignScript.Geometry import *
```

```
# Place your code below this line
```

```
#Definir variáveis e inputs
```

```
listresi = IN[0]
n=len(listresi)
materiaisdisp = IN[1]
valorin = IN[2]
erro="ATENCAO: betao inferior a C12/15"
```

```
#Protecção de erros
```

```
if valorin < 10:
    OUT= str(erro)
```

```
#Filtragem da lista introduzida para valores menores que “valorin”
```

```
#Procura do index do “match” na lista provisória
```

```
if valorin >=10:
    filter1 = sorted (i for i in listresi if i <= valorin)
    valorfin = max(filter1)
    index=filter1.index(valorfin)
```

```
#Output da classe do betão
```

```
OUT = materiaisdisp[index]
```


ANEXO 3

Objetivo: Avaliação dos resultados de ensaio e calculo da índice esclerométrico.

Linguagem de programação: *Python*.

```
# Enable Python support and load DesignScript library
import clr
clr.AddReference("RevitAPI")
from Autodesk.Revit.DB import *
clr.AddReference("RevitServices")
import RevitServices
from RevitServices.Persistence import DocumentManager
from RevitServices.Transactions import TransactionManager
doc = DocumentManager.Instance.CurrentDBDocument
```

#Criação das funções de mediana, média e desvio padrão

```
def median(lst):
    n = len(lst)
    if n < 1:
        return None
    if n % 2 == 1:
        return sorted(lst)[n//2]
    else:
        return sum(sorted(lst)[n//2-1:n//2+1])/2.0

def mean(data):
    n = len(data)
    if n < 1:
        raise ValueError('mean requires at least one data point')
    return sum(data)/float(n) # in Python 2 use sum(data)/float(n)

def _ss(data):
    c = mean(data)
    ss = sum((x-c)**2 for x in data)
    return ss

def stddev(data, ddof=1):
    n = len(data)
    if n < 2:
        raise ValueError('variance requires at least two data points')
    ss = _ss(data)
    pvar = ss/(n-ddof)
    return pvar**0.5
```

Place your code below this line

#Definir Variáveis

```
lista = IN[0]
length= len(lista)
menor=min(lista)
mediana=median(lista)
contador=0
contadortotal=0
percent=0
fr=0
df=[0] * length
avg=0
avglista=0
desvpadrao=0
```

#Contador da diferença entre as leituras e a mediana superior a 6

```
for x in range(0, length):
    if abs(lista[x]-mediana)>6:
        contador = contador + 1
        contadortotal=contadortotal+1
    if abs(lista[x]-mediana)<=6:
        contadortotal=contadortotal+1
```

#Calcula a percentagem

```
percent=float(contador) / (length)
percent=percent*100
```

#Divulgação do resultado

```
if percent > 20:
    percent="Conjunto de valores nao validos!!"
    OUT=str(percent)
if percent <= 20:
    OUT = median(lista)
```

ANEXO 4

Objetivo: Curva de calibração do esclerómetro de Schmidt.

Linguagem de programação: *Python*.

```
# Enable Python support and load DesignScript library
import clr
clr.AddReference("RevitAPI")
from Autodesk.Revit.DB import *
clr.AddReference("RevitServices")
import RevitServices
from RevitServices.Persistence import DocumentManager
from RevitServices.Transactions import TransactionManager
doc = DocumentManager.Instance.CurrentDBDocument
```

#Criação das funções de mediana, média e desvio padrão

```
def median(lst):
    n = len(lst)
    if n < 1:
        return None
    if n % 2 == 1:
        return sorted(lst)[n//2]
    else:
        return sum(sorted(lst)[n//2-1:n//2+1])/2.0

def mean(data):
    n = len(data)
    if n < 1:
        raise ValueError('mean requires at least one data point')
    return sum(data)/float(n) # in Python 2 use sum(data)/float(n)

def _ss(data):
    c = mean(data)
    ss = sum((x-c)**2 for x in data)
    return ss

def stddev(data, ddof=1):
    n = len(data)
    if n < 2:
        raise ValueError('variance requires at least two data points')
    ss = _ss(data)
    pvar = ss/(n-ddof)
    return pvar**0.5
```

Place your code below this line

#Definir Variáveis

```
medianas = IN[0]
provetes = IN[1]
length= len(medianas)
df=[0] * length
avg=0
desvpadrao=0
ajuste=0
```

#Cálculo da diferença entre a resistência de cada carote e a resistência dada pela relação base

```
for y in range(0, length):
    if medianas[y] >= 20 and medianas[y] < 24:
        df[y]=provetes[y]-(1.25*medianas[y] - 23)
    if medianas[y] >= 24 and medianas[y] <= 50:
        df[y]=provetes[y]-(1.73*medianas[y] - 34.5)
avg=mean(df)
desvpadrao=stddev(df, 1)
```

#Definição do K1

```
if length==9:
    k1=1.67
if length==10:
    k1=1.62
if length==11:
    k1=1.58
if length==12:
    k1=1.55
if length==13:
    k1=1.52
if length==14:
    k1=1.50
if length>=15:
    k1=1.48
```

#Cálculo do delta f

```
ajuste=avg - k1 * desvpadrao
```

#Divulgação do deslocamento

```
OUT = ajuste
```

ANEXO 6

Dynamo Player

< ↺

?

▶

Ensaio de Carotes e Provates

⚙️ ✎️

Run completed

✓

Select Model Element :

Select

Element : 282415

✓

File Path :

Browse...

Materials1.xlsx\..

✓

Pretende introduzir dados de ensaios de carotes ou provates? :

☒

True

✓

O valor introduzido diz respeito a carotes cilindricos? :

☒

True

✓

Valor 1: :

8.8

✓

Valor 2: :

9.7

✓

Valor 3: :

10.7

⚠️

Classe do Betão: :

ATENCAO: betao inferior a C12/15

Moradia.rvt